Ott./Nov./Dicembre 1988, Anno III n. 6
Mensile – Lire 7.000

La rivista per utenti di C-64/128 ed Amiga

COMMODORE GAZETTE

Più velocità per l'Amiga:

SCHEDE ACCELERATRICI E MEMORIA A 32 BIT

Programmare l'Amiga:

IL SISTEMA EXEC

CORSO DI LINGUAGGIO C

Prove software:

GEOCALC

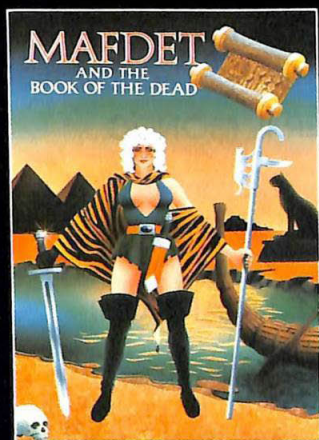
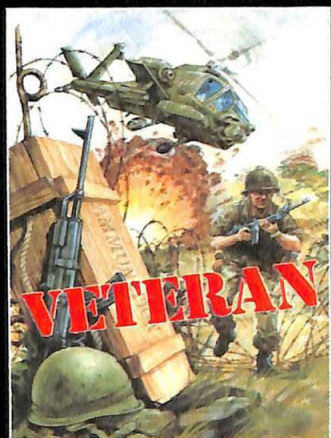
Il Clan dei pirati dello spazio seguaci del dio del male OKIRA, hanno rapito la vostra adorata pricipessa DIORA e stanno ora tenendo l'impero in ansia per il riscatto . . .

Voi dovete perlustrare tre pianeti per trovare la vostra principessa, e ciascuno di essi sarà fortemente scortato da guerrieri di OKIRA estremamente addestrati. Trova il pianeta in cui DIORA viene trattenuta, distruggi i pirati, e una fata ti csaudira un unico desiderio.

Amore e magia ti aiuteranno ad uscire vittorioso — tu non devi fallire questa missione. Raccogli punti di forza per ricaricare le tue armi e i sistemi di sopravvivenza in quanto tu combatti per aprirti un varco tra i pianeti.

VETERAN

La guerra è quasi terminata. La tua fazione sembra aver perso tutto. Ma c'è solo un'ultima possibilità per salvare la battaglia.

Tu devi lottare per portare a termine con successo 3 missioni mortali, abbattendo le posizioni nemiche, finchè tu non raggiungerai il Quartiere Generale dei nemici. Fai esplodere i serbatoi, le piazzole delle armi, e abbatti i gruppi dei soldati nemici. Le tue provviste sono limitate, per questo tu devi raccogliere lancia razzi ricambio e munizioni per la tua pistola, lasciati dai soldati in fuga.

La storia inzia nell'antico Egitto. Il Dio del male, SETH, ha rubato il Libro dei Morti e lo ha nascosto nel profondo dell'Inferno.

Tu sei Mafdet, il gatto Dio della Vendetta, il quale e stato convocato da ISIS (Dio della Terra) per recuperare il Libro e restituirlo al propolo Egiziano.

Raccogli amuleti che ti niuteranno nel tuo cammino; trasformati da gatto a umano per superare alcuni ostacoli; combatti per apirti un varco tra labirinti e prigioni; soprattutto — TROVA IL LIBRO.

ITAL VIDEO SRL CASTENASO (BOLOGNA)
TELEFONO 051 / 784010

Articoli

FOTO PATRICIA LEEDS

OTT./NOV./DICEMBRE 1988
Anno III, N.6

Rubriche

COMMODORE
GAZETTE

Una pubblicazione

Direttore Responsabile
Massimiliano M. Lisa

Redazione
Luca Giachino (capo servizio redazione tecnica)
Mauro Gaffo (redattore)
Nicolò Fontana Rava (servizi speciali)
Collaborazione Editoriale
Alfredo Macchi, Marco Menichelli, Sergio Fiorentini,
Franco Toldi, Fabio Rossi
Corrispondenti USA
William S. Freilich (sezione sviluppo)
Daniela D. Freilich (coordinamento generale)
Collaborazione Editoriale USA
Louis R. Wallace, Ervin Bobo,
Eugene P. Mortimore, Harriet Maybeck Tolly

Inviato speciale USA
Matthew Leeds

Segretaria di Redazione
Susan Visentin

Impaginazione e Grafica
Antonio Gaviraghi (capo servizio)
Andrea De Michelis
Fotografia
A.&G.,
Patricia Leeds
Disegni
Marco Piazza,
Giuseppe Festino

Direzione, Redazione, Amministrazione
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano

Fotocomposizione
IHT Gruppo Editoriale S.r.l. - Reparto grafica
Fotolito
C.S.T. S.r.l.
Via Melchiorre Gioia, 61 - 20124 Milano
Stampa
Eurograph S.p.A.
Via Oroboni, 38 - 20161 Milano

Distribuzione per l'Italia
Messaggerie Periodici S.p.A.
V.le Famagosta, 75 - 20142 Milano
Tel. 02/8467545

Distribuzione per l'estero
A.I.E. S.p.A.
Via Gadames, 89 - 20151 Milano
Tel. 02/3012200

Pubblicità
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano
Ufficio Pubblicitario IHT
Italia ed Estero 02/794181-799492-792612-794122
Fax 02/784021 - Telex 334261 IHT I

Servizio Abbonamenti

IHT Gruppo Editoriale
Servizio Abbonati
Via Monte Napoleone, 9 - 20121 Milano

SEGRETERIA ABBONAMENTI
Linea per registrazione abbonamenti
02/794181-799492-792612-794122

Commodore Gazette
Costo abbonamenti

Italia
12 numeri L. 84.000
24 numeri L. 168.000
36 numeri L. 252.000
Estero:
Europa L. 120.000 (10 numeri)
Americhe, Asia... L. 180.000 (10 numeri)

I versamenti devono essere indirizzati a:
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano
mediante emissione di assegno bancario o vaglia postale

Arretrati
Ogni numero arretrato: L. 14.000 (sped. compresa)

Autorizzazione alla Pubblicazione
Tribunale di Milano n. 623 del 21/12/85
Periodico Mensile - Sped. in abb. post. gr. III/70
ISSN: 0394-6991

La IHT Gruppo Editoriale è iscritta nel Registro
Nazionale della Stampa al n. 2148 vol. 22
foglio 377 in data 5/6/1987

Commodore Gazette è una pubblicazione
IHT Gruppo Editoriale

NOTE EDITORIALI

I primi giorni dello scorso novembre sono stati testimoni di una vera e propria epidemia. Negli Stati Uniti più di 6 mila computer sono stati infettati da un unico virus nel giro di pochissimo tempo.

Il creatore di questo agente patogeno si chiama Robert T. Morris jr., ed è uno studente universitario di 23 anni. Il programma era stato realizzato per vivere "innocentemente" all'interno di ARPANET, la rete computerizzata di telecomunicazione del dipartimento della Difesa statunitense, e passare lentamente da un computer all'altro. Ma un errore di progettazione ne ha causato una rapidissima riproduzione che è sfuggita al controllo del giovane programmatore. E così sono risultati infetti i computer collegati alla rete appartenenti all'Università di Berkeley, l'UCLA, la NASA, il MIT, lo Strategic Air Command, la National Security Agency... E l'ironia della sorte vuole che Robert Morris sia figlio del responsabile della sicurezza di ARPANET, proprio l'uomo che avrebbe dovuto rendere la rete immune da problemi di questo tipo.

Aldilà delle implicazioni padre/figlio va rilevato il dato: i network elettronici che distribuiscono la posta tra i computer sono vulnerabili al sabotaggio quanto lo è il servizio postale. Il pacco bomba dell'età dei computer è un programma che si duplica e che danneggia i computer che lo ricevono. Questi programmi distruttivi, i cosiddetti virus, possono consumare così tanta memoria che il computer che li riceve può rallentare i suoi processi al punto da diventare addirittura inutilizzabile.

Il virus di Morris ha sfruttato un errore nel software della rete per intrufolarsi e riprodursi. Ed è così che questi agenti patogeni si diffondono: sfruttano particolari *backdoor*.

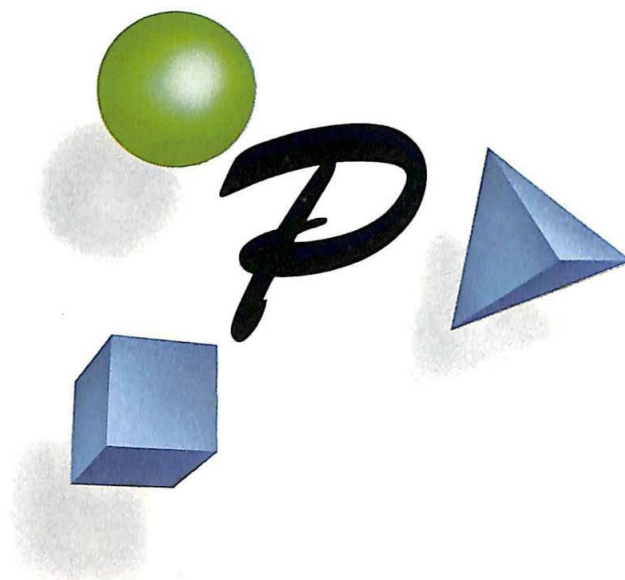
Si può fare qualcosa per prevenire il contagio? Una possibilità può essere quella di utilizzare programmi in grado di rilevare la presenza di eventuali virus. Ma anche così esiste sempre il rischio che il contagio venga rilevato dopo che ha già provocato danni irreparabili. Ci si può allora servire dei numerosi programmi antivirali presenti sul mercato, che hanno il ruolo di veri e propri vaccini. Ma i vaccini non possono far altro che indurre gli hacker a una continua sfida per cercare di aggirarli.

Il virus di cui abbiamo parlato era un programma altamente sofisticato da 60 mila byte, disegnato per decodificare le password e mascherarsi da utente legittimo. Era efficace solo su elaboratori Digital e Sun con sistema operativo Unix System 4.3. Ha causato la perdita di migliaia di ore/macchina di operatori tenuti lontani dai loro sistemi. Ma la sua conseguenza è stata in definitiva benigna. Ha messo in allerta i possessori di computer. È necessario prepararsi contro i futuri assalti! E questo vale anche per gli utenti di personal e home computer non collegati a reti telematiche, dal momento che i virus possono diffondersi anche attraverso i dischi.

Purtroppo non c'è nulla che garantisca una totale immunità, dal momento che, come abbiamo detto, qualsiasi antivirus è una nuova sfida per i programmatori. Del resto c'era da aspettarsela una vendetta da parte dei programmatori, per tutto il software copiato senza nessun guadagno per loro, i legittimi autori. Chissà fino a che punto si spingeranno...

ML

LA POSTA DELLA GAZETTE

LA VOCE DEI NOSTRI LETTORI

NELLA STANZA DI VETRO

Innanzi tutto le presentazioni, siamo il Levante Amiga Club (LEAC) composto da un gruppo di adulti appassionati di computer, per ora identificati nei sistemi Amiga e MS-DOS. Un giorno, decidendo di dedicarci in maniera seria e creativa al computer, cominciammo a scrivere e telefonare alle varie ditte che rientravano nei nostri interessi (Commodore, Atari e altre). Ebbene, nessuna risposta dalla Commodore mentre la Atari ha provveduto a rispondere prontamente, porgendoci anche una collaborazione attiva. Molti rivenditori (a parte qualche raro caso) si sono anche dimostrati impreparati e non informati, e non hanno disponibile ciò che propongono nelle pubblicità. Comunque, contavamo sullo SMAU per risolvere i problemi e quindi partimmo carichi di speranze e di domande...

Purtroppo gli espositori di accessori per computer Commodore erano assolutamente impreparati a rispondere a domande tecniche appena appena un po' più smaliziate e la Commodore in quanto tale era totalmente assente, o meglio presente al centro del mega-stand ove in una stanza trasparente i grandi capi e le procaci standiste beatamente stavano.

Di Amiga, a parte quelli predisposti per vecchi giochini, non ce n'erano... ah, sì, dimenticavamo quello sventrato nella bacheca! Ma le schede JANUS AT-XT, gli hard disk, i genlock e quanto più volte annunciato dalle riviste e nel nuovissimo catalogo... nulla c'era. In pratica si era delegata la presentazione della linea Amiga ai produttori esterni che brillavano per incompetenza («Sono solo due giorni che lavoro qui, non ne so niente» era la risposta tipo), incapacità di utilizzare quanto proposto, prototipi non funzionanti, senza poi parlare delle brillanti soluzioni espositive quali, per dirne una, l'unico monitor ad alta persistenza presente che mostrava un programma di totocalcio in bassissima risoluzione.

Il fiore all'occhiello era lo studio mobile AMIGA ben attrezzato, inaccessibile alla plebe, ma disponibilissimo per i megaclienti, dove dopo essere entrati con un bluff spudorato trovammo il tecnico tedesco addetto all'attrezzatura da sogno beato mentre copiava giochini con rampolli super raccomandati. Nonostante tutto provammo a fermare un addetto dell'"ufficio CBM". Dopo alcuni vani tentativi riuscimmo nell'impresa e alla nostra richiesta di un aiuto di qualsiasi genere nei confronti del LEAC ci rispose che la Commodore non ha tempo né personale da dedicare ai numerosissimi Commodore Club.

Il ritorno fu molto triste, ma le poche promesse fatteci d'immediata disponibilità di alcuni interessanti accessori (Videomaster...) ci aveva parzialmente confortato. Il giorno dopo, solito giro di telefonate e solite vaghe risposte. Nessuno conosce i nuovi prodotti pubblicizzati allo SMAU, o se sanno di cosa si tratta non si sa quando saranno disponibili. Morale: la maggior parte di quan-

Indirizzate tutta la corrispondenza per la rivista a:

COMMODORE GAZETTE
La posta della Gazette
Via Monte Napoleone, 9
20121 Milano

Preghiamo i lettori di essere concisi e concreti, per darci modo di rispondere al più grande numero possibile di lettere. La redazione si riserva comunque il diritto di sintetizzare le lettere troppo lunghe.

to viene pubblicizzato dalle riviste esiste solo su carta e si deve, quando è possibile, comprare a scatola chiusa. Molte volte c'è incompatibilità tra i prodotti, e chi li collauda è l'utente! D'altra parte non si capisce come campino certe ditte che promettono prodotti di cui non dispongono.

Spassosa situazione in un famoso Commodore Point dove, alla richiesta di un genlock, la risposta testuale è stata: «Non lo abbiamo, ci sembra che quel gioco non sia ancora arrivato...».

Levante Amiga Club
Via Terzi, 15
16039 Sestri Levante

SIDECAR E A2088

Vi scrivo queste poche righe sperando che possano essere utili ai possessori di A1000 e Sidecar. Se possedete il Sidecar costruito in Germania potete usare il software sviluppato per l'A2088 traendone i seguenti vantaggi:

1) nessun messaggio di debug viene inviato sulla porta seriale al momento dell'attivazione di PCWindow, sia mono, sia colore. Inoltre questi programmi si caricano e girano più velocemente

2) il Sidecar lavora con le espansioni di memoria e funziona correttamente anche quando l'Amiga lavora intensamente (prima, un intenso lavoro da parte dell'Amiga poteva portare a un arresto della parte PC)

3) il trasferimento dei file dal PC all'Amiga e viceversa avviene senza problemi

4) non ci sono più limitazioni per quanto riguarda il numero di partizioni Amiga sull'hard disk del PC (in precedenza quando il PC disk veniva riempito oltre il 60% finiva sulla partizione Amiga).

Per usufruire di questi vantaggi dovete procurarvi dal vostro rivenditore una copia del disco d'installazione dell'A2088. Quando ne sarete in possesso, leggete il file script chiamato "Sidecarinstall". Fate una copia del vostro *Workbench 1.2*, poi mandate in esecuzione questo script file. L'operazione crea un nuovo disco di boot per il Sidecar con tutto il nuovo software installato. Nota: non usate lo script file Bridgeinstall in quanto copierebbe il file LPT1 mentre il Sidecar necessita del file LPT1S. Se avevate una partizione sull'hard disk del PC dovrete riformattarla. Fate una copia dei dati contenuti nella partizione quindi riformattate e copiate tutti i dati nella nuova partizione. Potete ora rimettervi al lavoro con una macchina che vi sembrerà nuova!

Luca Vaghi
Cesano Maderno (MI)

TERRA DI NESSUNO

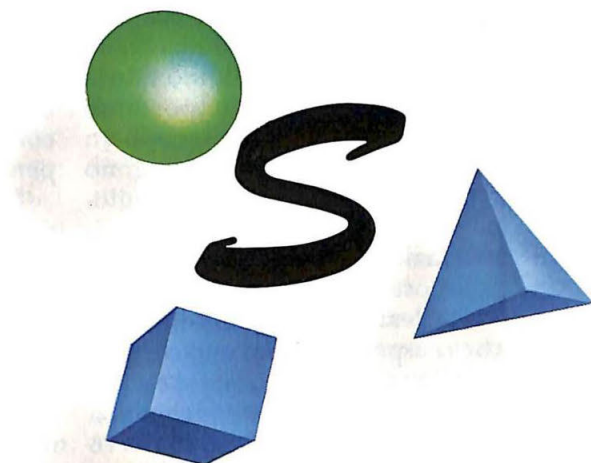
Sono un possessore di Amiga, pienamente soddisfatto di un computer le cui potenzialità – soprattutto in campo grafico – raggiungono risultati al di là di ogni previsione, ma nel contempo sono totalmente disgustato dell'incompetenza che regna sovrana nella rete di distribuzione e assistenza Commodore. Parlo per esperienza personale, dal momento che, fra l'altro, sono ancora in attesa dei manuali in italiano, per ottenere i quali ho spedito in raccomandata la fotocopia della garanzia. Adesso, a distanza di mesi, dopo aver inviato diverse lettere di sollecito, ho perso ogni legittima speranza. Anche perché, avendo avuto modo di ascoltare in merito altri amighisti, ho appurato di non essere stato il solo ad avere avuto delle delusioni. Purtroppo, da esperienze di questo genere escono gabbati non solo gli utenti Commodore, ma anche riviste professionali e ad alta tiratura quali, nella fattispecie, *Commodore Gazette*. Per non parlare poi degli elenchi di Commodore Point che vengono pubblicati su varie riviste: mi è capitato di fare parecchi chilometri per raggiungerne uno, debitamente pubblicato, per sentirmi poi dire dal gestore che non era abilitato alla vendita al minuto ma solo allo smercio all'ingrosso. Parallelamente, rammento di aver letto una dichiarazione del direttore delle vendite CBM, Albino Bertoletti, in cui si diceva che nel primo trimestre '88 sarebbero stati aperti mille Commodore Point in tutta Italia. Purtroppo, però, questo progetto non dev'essere stato attuato in modo completo, poiché in provincia di Pordenone e zone limitrofe, di Commodore Point non si è vista l'ombra.

Manca una fonte univoca nella Commodore Italiana, e in questa "terra di nessuno" esiste solo una gran congerie di notizie nebulose, a volte anche antitetiche. Aggiungo che prima di acquistare l'Amiga, avevo un Apple II GS e che l'uscita della nuova ROM rev. 01 mi è stata preannunciata con una raccomandata inviata dalla Apple al mio recapito (non ce ne sarebbe stato alcun bisogno, perché tutte le riviste ne avevano dato notizia senza alcun timore di smentita).

A questo punto capisco l'indecisione di molti potenziali acquirenti dell'Amiga, a tutto beneficio dell'altro 68000, cioè l'Atari; un computer dalle prestazioni magari inferiori, ma molto più assistito. È troppo comodo piazzare sul mercato una macchina potente e poi assumere un atteggiamento di disinteresse totale. I signori della Commodore Italiana devono capire che gli utenti necessitano di assistenza e chiarezza, anche perché penso che sia ingiusto che nelle vendite un computer eccellente come l'Amiga sia mortificato da una claudicante Commodore House.

Gastone Gava
Pordenone
■

SOFTWARE GALLERY

UNA GUIDA PER ORIENTARSI NEL MONDO DEL SOFTWARE

ULTIMA IV

Computer: C-64/128/Amiga
Versione: Amiga
Supporto: Disco
Prezzo: L. 39.000/49.000
Produzione: Origin Systems (Microprose)
Disponibile presso: Leader (Via Mazzini 15, 25020 Casciago – 0332/212255)

In queste colonne abbiamo spesso parlato di giochi di ruolo prendendo come pietra di paragone la serie di *Ultima* (con l'accento sulla "i") di Richard Garriott, e non è certo una scelta casuale. Ogni prodotto della serie, ormai arrivata al quinto capitolo, ha messo in mostra finora veri prodigi di fantasia e di qualità tecnica.

Vista la diffusione delle macchine della linea Amiga negli Stati Uniti, dov'è situata la sede centrale della Origin Systems di Richard Garriott, i programmatori della saga hanno cominciato da qualche tempo a convertire i loro lavori anche per questo computer, arrivando finalmente a presentare la meravigliosa "quarta parte" dell'epica vicenda sul nostro gioiello a 16 bit.

La grande attenzione ai particolari che contraddistingue questo programma si nota fin dal momento in cui si apre la pesante confezione: all'interno troviamo infatti oltre al disco del gioco ben due volumetti introduttivi scritti in un inglese medievaleggiante e decorati con rune elfiche, una fondamentale "Quick reference card" e infine una spettacolare mappa a colori stampata su stoffa. Ogni elemento della confezione è studiato in modo da creare un'atmosfera fantasy che riesce a coinvolgere il giocatore anche prima che il disco scivoli nel disk drive.

SCHEDA CRITICA

INSUFFICIENTE
Un pessimo prodotto che non merita nessuna considerazione.

MEDIOCRE
Il programma ha alcuni difetti di fondo, anche se nel complesso raggiunge quasi la sufficienza.

DISCRETO
Un prodotto accettabile, ma non aspettatevi grandissime emozioni.

BUONO
Raccomandato vivamente: tra i migliori programmi della sua categoria.

OTTIMO
Eccezionale! Fino ad oggi non si era mai visto niente del genere.

Ma veniamo al gioco. Dopo un rapido caricamento, appaiono i titoli e una finestra grafica nella quale possiamo avere un assaggio di quello che ci attende: in una grafica coloratissima ma semplice e funzionale, e con l'accompagnamento di una piacevole melodia medievale, vediamo infatti un gruppo di avventurieri alle prese con le forze del male che tentano di distruggere una città fortificata. La prima cosa da fare, a questo punto, è preparare un disco-scenario sfruttando la routine di copia presente nel primo menu. Dal momento che la procedura richiede circa cinque minuti, è bene sfruttare questo lasso di tempo per farsi un'idea dei comandi e della struttura del gioco.

Mentre la routine di copia fa il suo lavoro, apprendiamo dal manuale che i numerosi comandi vengono impartiti per la maggior parte da tastiera, e ogni tasto corrisponde a un'azione diversa. Nonostante la loro quantità, la scelta degli abbinamenti è logica e

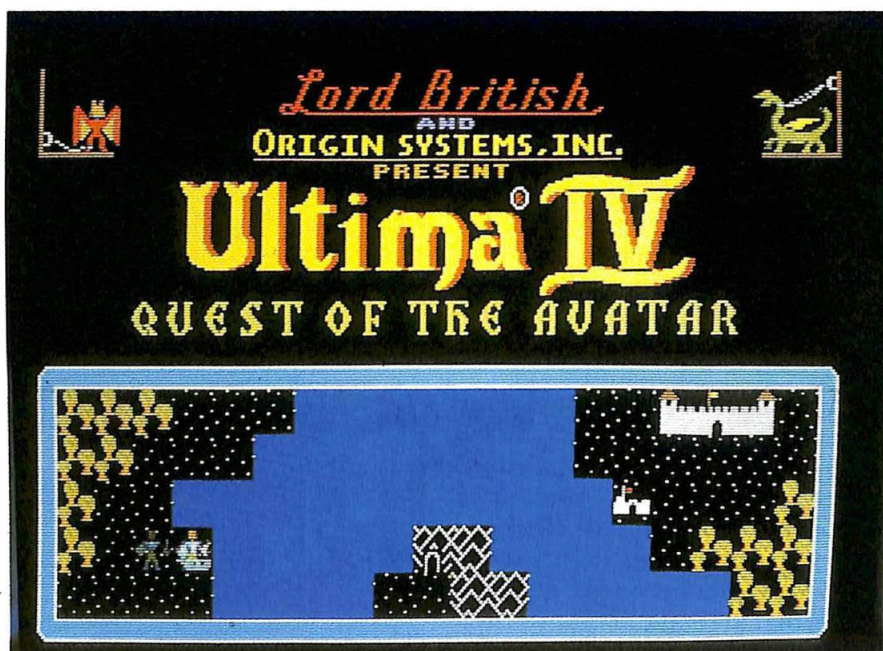
quindi facile da memorizzare ("A" sta per Attack, "B" per "Board"...). Dai manuali impariamo anche che la situazione in cui il nostro personaggio si trova non è delle più felici: ha il compito di liberare il mondo di Britannia dalle orde del Male, capitanate questa volta da una creatura demoniaca e sconosciuta. Si può sconfiggere una simile minaccia soltanto se si raggiunge la condizione di "Avatar", ovvero la purezza di spirito. Il compito del giocatore consiste innanzi tutto nel conquistare questo status, che gli permetterà di penetrare nel luogo da cui proviene tutto il Male, l'Abisso, per fermare il misterioso avversario.

Terminata la routine di preparazione dello scenario, il gioco prosegue chiedendo al giocatore d'identificarsi e presentando una serie di schermate di qualità non particolarmente elevata che raccontano come siamo penetrati tramite una porta dimensionale nel mondo di Britannia, e il nostro incontro con una misteriosa zingara. In una sequenza fondamentale per lo sviluppo del gioco, la zingara chiede al giocatore di fare alcune difficili scelte etico-morali di cui registra poi i risultati su una sorta di pallottoliere. La disposizione delle biglie del pallottoliere alla fine della seduta definisce il carattere e la professione del giocatore, che si materializza poco dopo al di fuori delle mura del castello di British, la capitale di Britannia.

Da questo momento in poi la struttura dello schermo cambia in maniera permanente. Nella grande finestra grafica di sinistra si vede una rappresentazione schematica del mondo che ci circonda, mentre la sezione di destra è riservata alle informazioni riguardanti il nostro personaggio e gli esiti delle nostre iniziative.

Come in ogni gioco di ruolo che si rispetti, il nostro personaggio è identificato da una serie di attributi personali che mostrano sotto forma numerica le sue doti fisiche e mentali e il suo bagaglio di esperienza: quest'ultimo valore è particolarmente importante perché soltanto raggiungendo un alto livello d'esperienza vengono rivelati alcuni particolari del gioco e diventano possibili determinate azioni.

Il nostro primo compito è trovare alleati per la difficile impresa che ci aspetta: li possiamo cercare in ognuna delle numerose città che costellano l'immensa mappa principale di Britannia (332.776 tene di personaggi sparsi per diverse città e villaggi si arriva dopo qualche tempo a identificare i nostri possibili compagni d'avventura, ma il problema ora è convincerli delle nostre qualità perché si uniscano al gruppo!

Una volta riunita – in un modo o nell'altro – una compagnia di otto avventurieri, ognuno dotato di caratteristiche e capacità diverse, inizia la ricerca della purezza interiore. Un'impresa non certo facile in un mondo nel quale si viene assaliti da mostruose abominazioni a ogni piè sospinto, ma

locazioni possibili!), ma le cose non sono semplici come sembra. Entrando nelle città vengono infatti richiamate dettagliatissime mappe secondarie popolate – ovviamente – dagli abitanti della città. Il guaio è che in *Ultima* ogni città ha almeno un centinaio di abitanti! Ognuno è in grado d'interagire con il giocatore, che si trova a dialogare con personaggi diversissimi, dal bambino cleptomane al povero storpio che chiede la carità cercando di evitare le guardie. Seguendo complicate ca-

grazie alla collaborazione dei nostri compagni si riesce a raggiungerla in sole due o tre settimane di gioco. Ma cosa sono due settimane, per un vero giocatore? Per ottenere l'illuminazione sono necessarie lunghe indagini fra gli abitanti delle città: l'unica via per raggiungere i mantra e le rune necessari per accedere ai templi (nascosti pure quelli!) dove si medita sulle virtù. Queste indagini portano il gruppo di avventurieri attraverso tutto il continente, obbligandoli a continui com-

battimenti con le forze del Male. Nelle fasi di scontro, lo schermo cambia per mostrare una visuale molto particolareggiata dell'area della battaglia, in cui ogni singolo personaggio può sfruttare le sue capacità tattiche, belliche o magiche per eliminare gli avversari. I nemici, che si presentano sotto una quarantina di forme diverse, crescono in numero e potenza col procedere del gioco creando una sorta di limite di tempo per i progressi del gruppo, ma con la loro forza cresce anche il valore delle ricchezze che portano con

sé. Questi tesori restano abbandonati sul terreno come preda per il vincitore dei combattimenti, e impossessandosene i giocatori possono finanziare i loro viaggi sempre più dispendiosi e l'acquisto di armi.

Inframmezzato da decine di battaglie, il gioco procede poi con la sfida degli immancabili dungeon, labirinti sotterranei particolarmente complessi che in questo gioco si sviluppano sino a una decina di livelli di profondità fornendo altre centinaia di possibili locazioni. Sul fondo di ogni dungeon si trova un altare, sul quale vanno collocate pietre magiche per penetrare nell'Abisso. In queste fasi sotterranee la grafica cambia notevolmente: la visuale "a volo d'uccello" viene mutata infatti in una rappresentazione soggettiva.

Una volta completata anche questa fase sono ormai passati alcuni mesi dal primo caricamento del gioco, ma non per questo l'interesse diminuisce: la colossale sfida di *Ultima IV* prosegue infatti con la ricerca dei manufatti che permettono la localizzazione dell'Abisso. Tutte le avventure sono accompagnate da un aumentare costante delle difficoltà, alle quali si aggiunge ben presto un ulteriore pericolo: la virtù tanto duramente conquistata si rivela infatti alquanto instabile, e un avventuriero che si lascia prendere dall'ira e colpisce un avversario in fuga vede scomparire dal suo elenco le virtù dell'Onore, del Valore e della Compassione, ed è costretto a ricominciare da capo la ricerca dell'illuminazione.

A ogni aspetto del gioco è stata dedicata un'attenzione incredibile, a partire dalla caratterizzazione dei numerosissimi personaggi sino alle musiche medievaleggianti che accompagnano lo svolgersi di ogni azione, e non ci si annoia nemmeno dopo mesi e mesi di gioco (parliamo per esperienza acquisita su altri computer). La varietà delle situazioni è tale che tutti troveranno elementi di loro interesse, dagli amanti dei combattimenti agli appassionati di magia, che in questo gioco hanno a disposizione una trentina d'incantesimi. La grafica, semplice ma non per questo approssimativa, è vivacemente colorata e proprio per la sua essenzialità riesce a rappresentare con chiarezza le situazioni più complesse. Il sonoro invece è decisamente di buona qualità, senza nessuna riserva, sia per quanto riguarda gli effetti sia per le musiche.

Il sistema di controllo è semplice e razionale, e permette di calarsi completamente nel gioco sin dalle prime fasi: la sua apparente limitatezza si traduce invece in una linearità che rende inutile l'uso del manuale, a differenza di quanto accade in numerosi altri RPG.

L'aspetto più caratteristico di *Ultima IV* è senz'altro la sua complessità, grazie alla quale Britannia diventa un mondo vero e proprio, privo di tutte le limitazioni di altri ambienti digitali. Le scorribande dei nostri eroi a cavallo, per mare e persino in mongolfiera permettono presto di apprezzare la grandiosità della mappa, e il numero di situazioni possibili è paragonabile soltanto alla sua estensione.

Ultima IV è un prodotto assolutamente unico, senz'altro il migliore del suo genere. Per gli appassionati non c'è nemmeno bisogno di consigliare l'acquisto. Chi non conosce i giochi di ruolo, invece, non potrà trovare esempio migliore di *Ultima IV*. **F.R.**

ULTIMA V

Computer: C-64/128
Supporto: Disco
Prezzo: L. 49.000
Produzione: Origin (Microprose)
Disponibile presso: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Otto anni fa, il primo *Ultima* di Richard Garriott, nella versione per l'Apple, segnava l'inizio dell'era dei giochi di ruolo su computer: acclamato come miglior programma dell'anno dall'autorevole rivista *Electronic Games* (edizione americana), il programma ebbe due seguiti sempre più perfezionati e coinvolgenti, che prepararono il mercato al grande successo di *Ultima IV*. Quando è giunta notizia dell'imminente pubblicazione di *Ultima V*, l'entusiasmo dei recensori è stato tale da far registrare i primi casi di "prenotazione": tutti volevano avere l'occasione di essere i primi a vederlo.

Il sottotitolo del gioco, "Un'avventura di proporzioni epiche", la dice lunga sul suo contenuto: il programma si estende su ben otto facciate di disco, sulle quali è contenuto il mondo già presentato in *Ultima IV* e un intero continente sotterraneo, il tutto rappresentato con una definizione grafica doppia rispetto a quella della versione precedente. Come apprendiamo dalla lettura del manuale e dall'introduzione che compare sullo schermo, siamo stati ancora una volta trasportati nel continente extradimensionale di Britannia, dove la pace seguita alla nostra missione di Avatar (si veda la recensione di *Ultima IV* per ulteriori chiarimenti) è solo un ricordo.

La scoperta di un intero mondo sotterraneo ha spinto infatti il buon sovrano Lord British a partecipare a un'esplorazione, dalla quale non è tornato. La scomparsa di Lord British ha dato modo al perfido Lord Blackthorne d'instaurare un cupo regime dittatoriale, che mantiene con l'ausilio dei neri e misteriosi Signori delle Ombre. Questi tetri ambasciatori individuano subito il nostro arrivo, e dopo aver colpito quasi mortalmente Shamino, il nostro compagno di viaggio, scompaiono nel nulla.

L'avventura vera e propria inizia nella capanna di Iolo, dove si può per la prima volta ammirare la nuova grafica del gioco. Subito ci si accorge però che qualcosa non quadra: dov'è andata a finire la bella colonna sonora che accompagnava le nostre gesta di aspiranti Avatar? Un'angosciata ricerca compiuta sulla guida al gioco composta da un fantomatico Dr. Cat (il programmatore che si è occupato della conversione per il C-64) ci spiega che le 16 musiche del gioco possono essere ascoltate solo con un C-128. Questo vuol forse dire che la memoria del C-64 è stata riempita al punto di non poter più gestire nemmeno una musica?

Il manuale (quello di Dr. Cat) fa alcuni vaghi accenni in questo senso, ma il fatto che siano stati necessari parecchi tentativi prima di far funzionare la routine di caricamento del gioco inizia a far sospettare che questo dottor micione non sia poi un granché come programmatore...

Una volta rappezzate nel migliore dei modi le ferite di Shamino, possiamo finalmente uscire dalla capanna per scoprire l'orticello del bardo, un pozzo dei desideri e una grafica generalmente ottima, che passa a un diverso livello di definizione quando ci allontaniamo dalla casetta e dobbiamo inserire il disco contenente la mappa principale. I comandi, rimasti quasi completamente invariati da *Ultima IV*, si rivelano molto pratici, ma al primo combattimento ci si rende conto di un'altra stranezza: a differenza di quanto accadeva nel gioco precedente, non controlliamo ciclicamente tutti i membri del nostro gruppo. Il computer sembra passarcene il controllo in maniera casuale. Alcuni rimangono fermi, altri giocano anche quattro turni consecutivi, e il

solito esame del manuale di Dr. Cat (quello di Lord British, relativo alla versione per l'Apple, non ipotizza nemmeno lontanamente simili problemi) non ci aiuta per nulla. L'unico comando che può darci il controllo di un personaggio in particolare ha anche l'effetto di bloccare completamente tutti i nostri avventurieri, lasciandoli vittime di mostri che (li abbiamo contati con le mani fra i capelli!) sferrano 32 attacchi per ogni turno di un nostro personaggio. Imputando la strana circostanza alla nostra scarsa familiarità con i comandi, proseguiamo l'avventura fino a giungere a una città, sempre nel silenzio più assoluto del nostro frustratissimo SID.

Qui la definizione grafica ritorna ottima, e l'atmosfera di cupa oppressione che si respira è davvero ben costruita: i continui pedinamenti delle guardie, gli editti di Blackthorne, il vagare minaccioso dei Signori delle Ombre, le domande trabocchetto che ci rivolgono i cittadini scontenti, contribuiscono a renderla sempre più realistica. I dialoghi sono più evoluti rispetto a quelli di *Ultima IV*, ma in compenso ci sono meno personaggi. Abbiamo appena il tempo di notare lo spaventoso rincarare delle merci (praticamente impossibile comprare un'arma, nonostante i nostri sforzi di avventurieri provetti) che ci si presenta un nuovo irritante problema: lo schermo si stringe!

Il calare improvviso della notte riduce drasticamente il nostro campo visivo, mentre le porte della città vengono chiuse e tutti vanno a dormire. Detto così potrà sembrare realistico, ma il fatto che i nostri personaggi non si stanchino minimamente e che durante la notte ci si trovi spesso rinchiusi da qualche parte rovina tutto e ci obbliga a passare parecchi minuti col dito premuto sul tasto Pass, nell'attesa che sorga il sole. Ovviamente, Lord British (il programmatore, non il suo omonimo digitale) aveva pensato anche a un comando da usare in questa evenienza, ma il geniale Dr. Cat ha dimenticato d'inserirlo tra le funzioni della macchina... anche se si è ricordato di elencarlo tra i comandi possibili.

Una volta tornata la luce, è il momento di tornare all'aria aperta e continuare il nostro giro turistico: fra una città e l'altra incontriamo le solite decine di mostri, che questa volta sembrano apparire casualmente. Se in qualsiasi altro gioco di ruolo la potenza dei nemici è proporzionale alla nostra, infatti, questa versione di *Ultima V* è capace di scagliarci addosso sin dal primo incontro uno stormo di draghi o un gruppo di Signori delle Ombre, che uniti alla difficoltà di gestione dei combattimenti conducono quasi sempre a una fine prematura del gioco (che oltretutto non salva la situazione automaticamente come faceva *Ultima IV*). Una soluzione ovvia in casi come questi è la magia, ma scopriamo ben presto che la sua efficacia è totalmente casuale e perdipiù identifica la posizione del gruppo agli occhi dei Signori delle Ombre, senza lasciarci praticamente scampo. Probabilmente la soluzione sta nell'uso corretto dei misteriosi oggetti presenti nell'ultima pagina del nostro inventario, che però non dovrebbero nemmeno esistere, secondo il manuale. Ci sarebbero ancora molte altre "stranezze" di cui parlare, come il fatto che a volte il gioco comincia a caricare e non si ferma più, o che non sembra esserci modo di guadagnare in alcun modo somme di un certo livello che ci permettano di uscire dall'avvilente ruolo dell'accattone, ma è meglio calare un velo pietoso sullo scempio perpetrato da Dr. Cat su questo capolavoro. L'atmosfera o la grafica non bastano a salvare un gioco frustrante e quasi ingiocabile, e gli unici aggettivi adatti a definire questo programma sono "deludente" e "irritante".

Concludiamo la recensione con l'amaro in bocca e la speranza che la versione per l'Amiga venga realizzata in modo degno. **F.R.**

STARFLEET I

Computer: Amiga
Supporto: Disco
Prezzo: L. 45.000
Produzione: Intersteel
Distribuzione: C.T.O. (Via Piemonte 7/F, 40069 Zola Predosa – 051/753133)

«Diario del Capitano, data astrale 118 e 46.5...»: con parole similari si apre ogni puntata della fortunatissima serie televisiva *Star Trek*.

Dopo quattro film, un numero spropositato di repliche televisive e una nuova serie di telefilm attualmente in programmazione negli Stati Uniti, il mito della missione quinquennale dell'Enterprise, la nave spaziale più famosa della galassia, non sembra ancora tramontare. L'equipaggio comandato dal capitano J. T. Kirk è già stato protagonista di due videogiochi arcade, ma questo programma offre possibilità ben più grandi di una semplice battaglia tra piloti dal grilletto facile: non vi era mai capitato di sperare di potervi arruolare un giorno nella Federazione?

Sia chiaro, *Starfleet I* non è ufficialmente correlato in alcun modo con la serie *Star Trek*, ma il programma parla indubbiamente della stessa Federazione che mandava in giro per la galassia Spock e compagni. Nonostante i nemici si chiamino "Krellans" invece che "Klingons", infatti, le partite sono giocate sulla base di ordini come «Curvatura due punto sei, signor Sulu!» e «Attivi i faser, signor Scott!».

Dopo una presentazione adeguatamente epica, accompagnata persino dalla *Cavalcata delle valchirie*, il giocatore deve inserire il suo nome e la parola d'ordine con la quale viene identificato dai computer della Federazione; questa prima procedura impedirà l'accesso al file da parte di eventuali altri giocatori. È un particolare molto importante, dal momento che la maggior parte del fascino di *Starfleet I* sta proprio nella competizione fra i giocatori, che lottano l'uno contro l'altro tentando di scalare la gerarchia militare nel minor tempo possibile. Addirittura il programma tiene conto di ogni progresso dei giocatori mediante precise statistiche e una lista dettagliata delle battaglie intraprese. Una volta portato a termine un certo numero di battaglie di un adeguato quoziente di difficoltà, la Federazione premia i vari giocatori con promozioni, medaglie e menzioni al valore, tutte accuratamente registrate in un file accessibile a tutti gli utenti del programma.

La strada verso l'alto ammiragliato, tuttavia, non è delle più facili: oltre alle battaglie, di cui ci occuperemo in seguito, il giocatore deve sopportare infatti anche alcune prove piuttosto pesanti... come la lettura del manuale di gioco per esempio. Il manuale di un centinaio di pagine (duecento, con la traduzione italiana) si dimostra una lettura essenziale per poter affrontare una missione con qualche speranza di successo, e solo dopo averlo assimilato completamente sarà possibile nutrire la speranza di far carriera nella Federazione.

Dopo la procedura d'identificazione, il giocatore può comunque scegliere il livello e la durata della propria missione. La scelta è assolutamente libera, ma soltanto una missione di livello almeno uguale a quello raggiunto fino a quel momento dal giocatore (ce ne sono dieci) viene considerata valida ai fini della carriera.

La Federazione provvede quindi a trasferire il giocatore e la sua nave – scelta in precedenza – nella zona di spazio in cui avverrà il combattimento. Le missioni di *Starfleet I* sono infatti invariabilmente di genere bellico, e in particolare riguardano la difesa di basi stellari.

Il gioco è ambientato in una regione stellare i cui quadranti sono indicati e descritti da terne di cifre che compaiono di volta in volta in un'apposita finestra. Ogni quadrante, a sua volta suddiviso in cento settori di cui un'altra finestra fornisce una visione d'insieme, contiene basi stellari (da difendere e da utilizza-

re come stazioni di rifornimento), stelle (da evitare o da sfruttare come scudi durante le battaglie) e nemici (da ridurre in polvere cosmica).

I nemici sono di due tipi: i crudeli Krellans, votati alla distruzione, e i più infidi Zaldrons, che sfruttano la loro avanzatissima tecnologia per schermare le navi da battaglia e renderle così virtualmente invisibili; entrambi vorrebbero distruggere le basi stellari confederate sparse nella regione. Da parte sua, il giocatore deve tentare di distruggere almeno il 75 per cento degli alieni entro un dato limite di tempo, utilizzando i potenti mezzi di cui è fornita la sua nave.

Il gioco si svolge in turni scanditi dalle mosse del giocatore, che agisce selezionando i possibili comandi dal menu nella parte bassa dello schermo. Possono essere schematicamente divisi in tre categorie: comandi di navigazione, di esplorazione e di attacco/difesa.

I comandi di navigazione permettono al giocatore di dirigere la nave confederata ovunque, ma è necessario tener conto del fatto che inizialmente non si conoscono la posizione e la consistenza del nemico, né le coordinate delle stelle e delle nostre basi.

Per far fronte a questa necessità sono presenti i comandi di esplorazione, che mettono a disposizione del giocatore un certo numero di sonde da lanciare nelle regioni ancora incognite dello spazio, identificando così il contenuto dei vari quadranti.

L'ultimo set di controlli riguarda i comandi di attacco/difesa, che governano torpedini spaziali, mine, scudi e raggi trattori. Utilizzandoli, il giocatore decrementa la quantità di energia a sua disposizione ma ha la possibilità di eliminare i nemici e di prendere prigionieri fra le forze aliene.

Starfleet I è basato sull'attenta amministrazione delle energie di bordo, e permette quindi al giocatore di sapere di volta in volta quanta energia viene sprecata da ogni singolo colpo e da ognuno dei quattro scudi di cui è fornita la nave. L'energia persa viene reintegrata attraccando alle basi stellari (dove vengono anche riparati tutti i nostri danni), ma questa procedura richiede tempo, per cui spesso si è costretti a giocare mezza partita con un'astronave ridotta a brandelli pur di fermare gli attacchi nemici. Durante le battaglie, che ai livelli più alti possono diventare particolarmente complesse, sono richieste notevoli doti tattiche e di calcolo, ma la giocabilità del programma rimane comunque alta in ogni sua fase.

Le opzioni a disposizione del giocatore sono molte e tutte utili, dalla ricerca di eventuali infiltrati alieni a bordo, all'autodistruzione della nave in caso di situazioni particolarmente disperate. Il gioco, tutto sommato piuttosto semplice come concezione, diventa molto impegnativo ai livelli più alti che purtroppo comportano anche missioni lunghe e, dopo qualche tempo, monotone. Nonostante il poderoso manuale, infatti, il gioco non riesce a creare un universo che coinvolga totalmente il giocatore e rimane sempre un freddo e arido simulatore. Questo non vuol dire certo che *Starfleet I* non abbia un suo fascino: l'ambiente militaresco della Federazione è stato ricreato molto bene, e lo spirito di competizione che riesce a creare in gruppi di giocatori abbastanza numerosi è notevole. L'idea di mettere a disposizione degli utenti un file che contiene i progressi dei vari giocatori è eccellente, e inoltre la Interstel fornisce nel manuale tutti i dati necessari per "truccarlo", strabiliando i vostri amici.

Starfleet I è, in definitiva, un programma essenziale per tutti gli appassionati di *Star Trek*, ma come molti altri prodotti legati ai serial difficilmente entusiasmerà il giocatore casuale. Può comunque essere considerato una semplice e piacevole introduzione al mondo dei war game, o una versione particolarmente evoluta della vecchia battaglia navale. Un bel salto: dai fogli di quaderno al monitor dell'Amiga.

F.R.

BOBO

Computer: Amiga
Supporto: Disco
Prezzo: L. 40.000
Produzione: Infogrames
Distribuzione: C.T.O. (Via Piemonte 7/F, 40069 Zola Predosa – 051/753133)

La Infogrames è per molti un sinonimo di qualità. Da *Vera Cruz* a *Murder on the Atlantic*, da *The Sidney Affair* al recentissimo *Passeggeri del vento*, ogni gioco della software house francese è stato un avvenimento che ha offerto molte serate di divertimento. Ecco perché ora, di fronte a questa nuova produzione, ci è veramente difficile non manifestare subito un profondo senso di delusione.

Mettiamo comunque da parte le questioni di cuore e vediamo qual è l'intreccio del gioco. Bobo, il personaggio con cui il giocatore è invitato a identificarsi, è un infaticabile detenuto del carcere di Inzepocket. Il nostro eroe è quindi impegnato nelle più diverse mansioni carcerarie, e ciascun momento della sua giornata è stato trasformato dai programmatori della Infogrames in un minigioco. In tutto ci sono sei "imprese quotidiane" in cui il giocatore può cimentarsi: passiamole rapidamente in rassegna.

Innanzitutto la mensa. Su due squallide panche sei detenuti attendono impazientemente che Bobo serva loro il rancio. Agendo senza sosta sul joystick il giocatore dovrà cercare di soddisfare velocemente i loro desideri; in caso contrario finirà inevitabilmente al suolo ricoperto di brodaglie e avanzi vari.

Sempre di argomento alimentare, il secondo gioco porta Bobo nelle cucine del carcere per la corvè pela-patate. Si tratta di un compito davvero titanico e dovremo sfoderare tutta la nostra bravura per prendere il giusto ritmo e non finire sepolti sotto una montagna di tuberi in rapida crescita.

La vita di un detenuto in un carcere comporta ripetute corvè di lavaggio dei pavimenti, almeno a Inzepocket. Anche qui la rapidità è essenziale e Bobo dovrà lavorare velocemente di ramazza tra le gambe degli altri carcerati, per evitare le punizioni che seguono l'ispezione.

Che carcere sarebbe mai Inzepocket se i suoi ospiti non tentassero qualche volta l'evasione? Ma dal momento che sono degli allegri giocherelloni non si sognano neanche di scavare lugubri gallerie, e tentano la fuga semplicemente saltando dalla finestra della cella su un trampolino manovrato da Bobo. Un salto azzeccato, e il rimbalzo porterà il compagno di turno oltre il muro di cinta. Ma quando verrà il turno del povero Bobo?

Evidentemente qualcuno si impietosisce, perché nel gioco successivo Bobo è fuori dal carcere e corre sui fili della luce, saltando da un filo all'altro per evitare le scariche elettriche. Ma se anche gli va bene, il destino di Bobo è comunque quello di essere sempre ricatturato, e infatti l'ultimo gioco lo vede alle prese con il rumorosissimo dormitorio del carcere. Se non riesce a calmare i compagni, cullandoli, è destinato a passare la notte in bianco.

L'idea di partenza, che ha ispirato i programmatori di *Bobo* non era cattiva: si trattava di cogliere il lato comico della vita carceraria, creando un sorridente personaggio che sapesse trasformare in gioco anche i momenti più duri della giornata. Una sorta di esorcizzazione del mondo "dietro le sbarre". L'ambientazione poteva essere suggestiva e surreale, visto

che il tono scanzonato non richiedeva certo una stretta aderenza con la realtà.

Ma qualcosa nella realizzazione del gioco dev'essere andata male e siamo certi che non erano questi i risultati a cui miravano i programmatori della casa francese. L'ambientazione è davvero molto povera, l'animazione procede a scatti e i giochi, presi uno per uno, peccano tutti di eccessiva semplicità. Dalla Infogrames, senza alcun dubbio, possiamo aspettarci di meglio.

F.T.

GAME OVER II

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Dinamic
Distribuzione: Lago (Via Napoleona 16, 22100 Como – 031/300174)

In un clima da saga interstellare continuano in *Game Over II* le avventure della bellissima dittatrice Gremla e del suo luogotenente Arkos. Ricordate la ribellione di Arkos e le sue scorribande sui pianeti Hypsis e Sckunn? Se avete apprezzato la varietà degli scenari e degli avversari da affrontare, e la velocità del gioco, in *Game Over II* sarete perfettamente a vostro agio.

Le cose tuttavia ora sono cambiate. Gremla è stata sconfitta, le truppe della fanteria siderale hanno invaso il Pianeta imperiale e ovunque si festeggia la fine del crudele Impero. Ma proprio Arkos, l'eroe della vicenda precedente, non arriva per i festeggiamenti. Invano vengono inviati messaggeri: le sue tracce sembrano essersi perdute. Finalmente emerge la tragica realtà: Arkos è stato catturato dagli eredi di Gremla, ben decisi a restaurare l'Impero caduto.

Il comandante Locke, il nuovo eroe, dovrà affrontare una serie di avventure per liberare il vecchio eroe ormai fuori dal centro dell'azione. Queste le premesse dell'avventura grafica con cui la Dinamic tenta di replicare il successo di *Game Over*. Vediamola ora nuovamente alla prova.

Come il suo predecessore, anche *Game Over II* è diviso in due parti e per accedere alla seconda, è necessario entrare in possesso di un codice rintracciabile solo nell'ultima scena della prima.

La prima parte del gioco si articola in quattro scene. Innanzitutto l'avvicinamento a Moon 4, un classico shoot'em up spaziale a scorrimento orizzontale, non troppo difficile. Quindi la ricognizione, con nuovi scontri aerei, attacchi missilistici e pericolose eruzioni vulcaniche.

Eccoci poi trasportati nel mondo sotterraneo del pianeta e nella sua grande palude per una delle fasi più godibili del gioco; se la superiamo riuscendo anche a impadronirci di un Adrec clonico, possiamo salirgli a cavalcioni e inoltrarci nella palude. Ancora qualche mostro da eliminare (ma che palude sarebbe, senza mostri?) e possiamo finalmente accedere alla seconda parte del gioco.

Inutile proseguire con l'esame delle successive scene del gioco, ormai tutti avranno capito cosa possono aspettarsi da *Game Over II*: combattimenti dopo combattimenti, nelle più svariate condizioni e con i più diversi avversari, una grafica accettabile, sprite ben definiti, una buona giocabilità. Quanto basta agli appassionati dei giochi d'azione per trascorrere piacevolmente alcune ore. Per coloro a cui i mostri e i combattimenti non bastano, la Dinamic ha provveduto a fornire sul retro del disco anche il primo gioco della saga. Se quest'idea prende piede, visto che le saghe ormai sono la norma, ci troveremo ad acquistare ogni volta due o tre giochi al prezzo di uno. **F.T.**

CAVEMAN UGH-LYMPICS

Computer: C-64/128
Supporto: Disco
Prezzo: L. 22.000
Produzione: Dynamix (Electronic Arts)
Distribuzione: C.T.O. (Via Piemonte 7/F, 40069 Zola Predosa – 051/753133)

La scena è di quelle che ogni possessore di Commodore 64 ha già visto almeno una volta: il tedoforo entra dalla parte destra dello schermo, accompagnato dalla notissima fanfara olimpica. Con poche rapide falcate raggiunge la base del braciere, appoggia la fiaccola all'ara, e in pochi istanti la fiamma olimpica si accende. Ma là dove i famosi giochi sportivi della Epyx si fermavano, questo programma prosegue mostrandoci il resto della vicenda: il cielo si oscura, l'atleta guarda preoccupato verso l'alto e... rimane spiaccicato dal titolo del gioco, che precipita dal cielo sotto forma di un enorme masso rozzamente scolpito. Benvenuti alle Olimpiadi della preistoria, le Ugh-limpiadi.

Caveman Ugh-lympics è un gioco decisamente fuori dalla norma: sin dall'esilarante sequenza iniziale, il giocatore capisce che in questo programma la serietà non ha diritto di cittadinanza. Il gioco si divide in un gruppo di sei gare, introdotte da una seconda cerimonia d'apertura e chiuse dalla rissosa premiazione degli atleti.

Prima di cominciare, il giocatore sceglie il proprio rappresentante in un gruppo di sei atleti dotati ognuno di abilità e handicap particolari. Prima dell'inizio delle gare, è possibile esaminare le schede di ogni personaggio (scritte in un incredibile inglese "preistorico"), dopodiché avviene la selezione... prendendo a colpi di clava l'icona corrispondente. Ogni personaggio ha le sue particolari abilità, fuorché Gronk, che sa far bene tutto, e Vincent che è un completo disastro; così dopo la prima partita resta la curiosità di vedere come se la caverebbe un altro atleta. Ogni cavernicolo sceglie poi uno sponsor che ne accompagnerà le gesta atletiche, e finalmente hanno inizio i giochi.

Le gare sono introdotte da una seconda cerimonia di apertura, ancora più comica della prima, nella quale il tedoforo è un ominide dalle movenze scimmiesche e la fiaccola si spegne prima dell'arrivo al braciere: è sicuramente una delle sequenze animate più spassose dell'intero gioco.

La specialità che inaugura le Ugh-limpiadi è il lancio della moglie: le regole sono quelle del più noto lancio del martello, ma qui l'attrezzo è sostituito dalle paffute consorti degli atleti: le riluttanti matrone vengono fatte ruotare sempre più velocemente, per essere poi sganciate in direzione di un prato sul quale ogni tre metri c'è un paletto indicatore piantato nel terreno. Una prima valutazione della nostra abilità può essere dedotta dall'eloquente mimica del proiettile umano, mentre le misure ufficiali vengono fornite poco dopo da giudici equipaggiati con l'ultimo grido degli strumenti di misura preistorici: un'attrezzatura digitale... così chiamata perché le misure vengono calcolate contando sulle dita.

La seconda prova, introdotta come le altre da un graffito sul quale è malamente inciso il nome della specialità, è la corsa sui dinosauri. Accovacciato sul dorso di un lucertolone molto poco entusiasta, il nostro atleta deve correre lungo una pista piena di ostacoli da saltare. Per controllare l'animale è possibile dare alla cavalcatura ben dosati colpi di tallone sui fianchi, ma il sistema più pratico è senz'altro prendere a clavate la testa del dinosauro. Questa semplice manovra ha effetti diversi a seconda della forza del colpo: un colpo leggero farà accelerare il dinosauro e uno più forte gli farà superare automaticamente diversi ostacoli alla massima velocità, ma se si esagera la povera bestia stramazza al suolo priva di sensi. Questa è l'unica gara rappresentata in split screen, conclusa come le altre dalla pre-

miazione dei migliori atleti con tanto di medaglie granitiche.

La gara successiva è rappresentata da una specialità degli uomini di Neanderthal: l'accensione del fuoco. Questa operazione apparentemente semplice è in realtà la più complessa di tutte le Ugh-limpiadi: per riuscire ad accendere il fuoco prima del nostro avversario serve anche parecchio colpo d'occhio oltre a una notevole velocità nello strofinare i bastoncini. Per attizzare il fuoco bisogna infatti soffiare a tempo debito sulle scintille, ma stando bene attenti a non soffiare troppo o troppo poco: riuscirci senza essere soffocati dal fumo, senza esaurire il fiato e senza essere bastonati dal proprio avversario è una vera impresa.

Per ritrovare il morale dopo le cocenti sconfitte della gara precedente, non c'è modo migliore che scaricare i nervi in una delle gare più spassose: il duello con le clave. Per dimostrare tuta la loro abilità, gli atleti ugh-limpici antenati degli schermidori di oggi devono affrontarsi su di un'alta e precaria piattaforma rocciosa, dalla quale è molto facile cadere. La gara inizia con una fase d'intimidazione che ci permette di spingere l'avversario verso il bordo della piattaforma: è sufficiente sballottare il più possibile il joystick per riuscirci. Poi si passa al duello vero e proprio, nel quale i duellanti metteranno in mostra un vasto repertorio di mosse più o meno legali, fra cui la famosa "Guarda l'uccellino!", rappresentate con una grafica da cartone animato. I due atleti saltellano quando vengono presi a bastonate sugli alluci, salutano i giocatori prima di lasciarsi cadere nel baratro e compiono altre prodezze degne dei migliori episodi di Wile E. Coyote. Per la vittoria si combatte al meglio dei tre incontri.

La specialità seguente è particolarmente sadica, e viene indicata come "Corsa della tigre dai denti a sciabola". Il particolare pathos nasce dal fatto che la tigre corre sì, ma dietro a due sfortunati atleti che arrancano sulle loro gambe tozze saltando terribili ostacoli come tappeti di cactus o pozze di pece, e voltandosi spesso a osservare terrorizzati il loro inseguitore. Oltre alle mosse comuni a tutti i giochi che prevedono una corsa, qui abbiamo anche la possibilità di afferrare il nostro avversario e gettarcelo alle spalle... e buon appetito alla tigre dai denti a sciabola.

Supponendo di aver raggiunto sani e salvi l'albero che segna la fine di questa drammatica gara, gli atleti passano all'ultima competizione: il terribile "Salto del dinosauro", che è un po' l'equivalente dell'odierno salto con l'asta. Come sempre i problemi non mancano. Questa volta sono rappresentati dal dinosauro da saltare: un gigantesco tirannosauro che attende a fauci spalancate un errore del nostro atleta. Conclusa anche questa prova, il gioco termina con la cerimonia di premiazione dei tre atleti migliori fra i sei partecipanti, che si risolve ovviamente in una colossale rissa a colpi di clava.

Nel corso del gioco l'elemento che colpisce di più è senz'altro la grafica, sia per le notevoli dimensioni degli sprite sia per l'animazione fluida e piena di umorismo che accompagna ogni gara. La parte sonora non è certamente all'altezza della grafica, ma la fantastica e immediata giocabilità bilancia egregiamente questa piccola pecca. Lo spirito "preistorico" del gioco non manca di farsi notare anche nelle situazioni più impensabili, e unito alle numerose opzioni di contorno (allenamento, multigiocatore e salvataggio dei record, palesemente ispirate alla serie sportiva Epyx) fa di *Caveman Ugh-Lympics* un piccolo capolavoro tra i giochi di genere sportivo. Probabilmente piacerà sia agli appassionati di giochi sportivi sia a quelli che li odiano e vogliono finalmente farsi quattro risate alle spalle dei fanatici della forma fisica. L'unica pecca, che però è il problema comune di tutti i giochi a carattere sportivo, è la ripetitività: a lungo andare ci si annoia anche del lancio della moglie, se la moglie è sempre la stessa.

F.R.

BUBBLE GHOST

Computer: C-64/128/Amiga
Versione: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 19.000/22.000/34.000
Produzione: Ere Informatique
Distribuzione: C.T.O. (Via Piemonte 7/F, 40069 Zola Predosa – 051/753133)

Il C-64 non finisce mai di stupire i suoi più affezionati utenti. Nonostante la sua età più che rispettabile, resiste ancora agli anni e ai cambiamenti imposti dalla rapidissima evoluzione dell'informatica senza perdere il passo.

Qualche volta però ci si stupisce anche per la ragione opposta... d'altra parte la parola "stupore" non implica necessariamente qualcosa di positivo. Il C-64 è una macchina strana: da un lato capita ancora che tra i molti programmi sfornati mese dopo mese dalle software house ce ne sia qualcuno in grado di farci scoprire capacità che ancora non sospettavamo nel nostro vecchio macinino a otto bit, e dall'altro si affiancano a questa produzione di eccellente qualità programmi quasi anacronistici, che dietro a una veste grafica appena sufficiente celano caratteristiche di giocabilità che ci riportano a un livello tecnico superato da anni.

Proprio a quest'ultima categoria, purtroppo, appartiene *Bubble Ghost*. Si tratta di un gioco d'azione nel quale l'obiettivo è riuscire a spingere una bolla di sapone, senza farla scoppiare, attraverso varie stanze.

Il protagonista dell'avventura è un fantasma che deve impostare con i suoi sbuffi la direzione della bolla di sapone, evitando che vada a toccare gli oggetti sparsi nelle stanze e facendole attraversare indenne il cunicolo di uscita. Dopo aver superato ogni stanza, si riceve un bonus per il tempo realizzato e ci si ritrova tutti e due – fantasma e bolla di sapone – in un'altra stanza dal percorso via via sempre più intricato.

Il fantasma viene controllato dal giocatore tramite joystick: premendo il tasto Fire e contemporaneamente muovendo la leva, si fa ruotare il fantasma finché non si sistema nella direzione migliore, poi ci si avvicina alla bolla e si preme nuovamente il tasto per soffiare. Più a lungo si tiene premuto il tasto, maggiore sarà l'intensità del soffio e quindi la velocità della bolla.

Se vi state chiedendo che altro c'è, mettetevi subito il cuore in pace: non c'è proprio altro. A parte il progressivo aumentare della difficoltà e la possibilità di soffiare su qualche oggetto per ottenere altri bonus (per esempio spegnendo le candele) non esiste nessuna variazione sul tema. Il gioco non è nemmeno difficile... Chi scrive, alla prima partita, senza aver letto le istruzioni e senza nemmeno impegnarsi troppo, ha superato tranquillamente le prime otto stanze prima di trovarsi di fronte a qualche difficoltà!

Si rivela quindi veramente difficile trovare anche i più piccoli motivi di emozione, visto che nemmeno la grafica si salva dalla monotonia che caratterizza tutto il gioco.

Per gli scettici e i malfidenti che volessero nonostante tutto provare il gioco e si aspettano qualche suggerimento strategico, consiglio di far muovere la bolla molto lentamente (anche se il punteggio si abbassa quando s'impiegano tempi maggiori). Nessuna stanza, infatti, è impossibile da superare, e andando avanti all'infinito potrete raggiungere qualunque punteggio. L'unico ostacolo davvero insuperabile è senza dubbio la noia.

Le critiche negative potrebbero anche continuare. Ma se penso a tutte le decine e decine di persone che hanno lavorato sul programma con impegno – magari per mesi – (o almeno così si spera) mi sento quasi dispiaciuto. D'altra parte il mercato del software è un po' come un concorso a premi... riprova e sarai più fortunato.

N.F.R.

BARBARIAN II

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Palace
Distribuzione: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Quando fu pubblicato circa un anno fa, il primo *Barbarian* della Palace fu uno dei videogiochi più chiacchierati e contestati della storia del C-64: era infatti uno dei primi esponenti del genere ultraviolento (che ebbe poi il suo massimo alfiere in *Deathwish III*) e rischiò di essere ritirato dal mercato, sia per la violenza gratuita e realistica che

proponeva sia per la campagna pubblicitaria basata sulla prosperosa figura di Maria Whittaker.

Dopo una storia legale piuttosto tormentata, *Barbarian* continuò a essere venduto senza problemi e le polemiche presto si affievolirono sino a scomparire. Quello che rimase fu invece l'entusiasmo dei giocatori alle prese con una sfida estremamente originale, realizzata oltretutto in maniera ammirevole ed efficace. Per questo secondo capitolo di avventure barbariche, la Palace non poteva certo limitarsi a proporre un altro duello fra spade celtiche, e infatti le differenze rispetto all'avventura d'esordio superano di gran lunga le somiglianze.

La prima schermata offre due possibilità di scelta per il proprio rappresentante sullo schermo: una mossa probabilmente studiata per mettere a tacere chi aveva additato *Barbarian* come un gioco maschilista. Le alternative sono il forzuto barbaro senza nome già protagonista del primo episodio e la formosa principessa Mariana.

Qualunque sia il personaggio scelto, il programma procede con un breve caricamento e con l'introduzione al primo livello, una pianura battuta dal vento. A differenza di *Barbarian I*, dove l'azione si svolgeva in uno scenario statico, questo secondo episodio viene giocato attraverso una serie di schermate a scroll orizzontale collegate da un sistema di tunnel, che formano un vero e proprio labirinto.

Vi sono in tutto quattro livelli di gioco, e ogni volta il compito del giocatore è quello di trovare e raccogliere i due oggetti che permettono di aprire il passaggio per il livello successivo. Gli ostacoli sono numerosi e si possono dividere in due categorie: gli ostacoli naturali e gli abitanti dei livelli.

Nel primo gruppo ricadono fiumi, macigni e pozzi senza fondo, ma tutti questi impedimenti possono essere evitati o superati con l'aiuto della bussola a forma di spada presente nella parte bassa dello schermo. Quelli che creano i veri problemi sono piuttosto i mostri che vagano lungo i diversi schermi.

Le malvagie e pericolose creature di cui stiamo parlando appaiono sotto numerose forme – strane galline mutanti, giganteschi draghi divoratori di teste... – ma con disegno e animazione sempre eccellenti (la stessa cosa vale anche per il nostro personaggio), e soprattutto con una diabolica intelligenza che può mettere in difficoltà il giocatore più esperto. A differenza dei nemici che s'incontrano nella maggior parte dei giochi di questo tipo, i mostri di *Barbarian II* sono particolarmente abili nell'evitare i nostri colpi e nell'attuare tecniche di "colpisci e fuggi" difficilissime da controbattere.

I combattimenti sono il punto di forza del gioco, che non è nemmeno dotato di musica proprio perché tutta la memoria del C-64 viene impiegata nell'animazione. Ogni colpo produce grandi schizzi di sangue e paurose mutilazioni, e la fine del combattimento è segnata da un'impressionante vampata verde seguita dalla scomparsa del cadavere.

Dopo una lunga serie di duelli a colpi d'ascia (o di spada, se il nostro personaggio è la principessa) e dopo avere attraversato caverne, sotterranei e un tempio, il giocatore raggiunge finalmente la sua meta: in una buia segreta langue in catene il corpo della compagna, o del compagno. Un potente colpo di lama taglia le catene, ma prima dell'*happy end* il giocatore si trova di fronte al suo più terribile nemico: il negromante Drax.

Il programma mette in mostra senza dubbio una grafica fra le migliori viste negli ultimi tempi, adeguatamente corredata da ottimi effetti sonori. Inoltre, la ricerca dei vari oggetti lungo l'intricato sistema di cunicoli sotterranei ne fa un prodotto interessante anche per l'appassionato di labirinti e avventure dinamiche, per quanto i violentissimi duelli costituiscano il 90 per cento del gioco.

Alcuni episodi davvero eccessivi, come per esempio la scena in cui un enorme sauropode stacca con un morso la testa del nostro personaggio, non possono certo far classificare *Barbarian II* come un gioco educativo, ma questo non fa più gridare nessuno allo scandalo, e soprattutto non impedisce a noi di segnalarlo come un buon programma.

F.R.

STARGLIDER 2

Computer: Amiga
Supporto: Disco
Prezzo: L. 49.000
Produzione: Argonaut Software (Rainbird)
Disponibile presso: Leader (Via Mazzini 15, 25020 Casciago – 0332/212255)

Per ogni home computer c'è almeno un programma che riesce a sfruttare al massimo le capacità della macchina, e diventa quindi un vero e proprio veicolo pubblicitario per l'hardware. Il VCS della Atari aveva *Space Invaders*, il BBC sbalordì tutti con il suo *Elite* e, più recentemente, l'Archimedes a 32 bit ha fatto di *Zarch* un vero e proprio onnipresente "demo".

Fino a oggi, però, era difficile trovare un titolo che ricoprisse lo stesso ruolo per il nostro beneamato Amiga. Nella moltitudine di prodotti software dedicati a questo computer esistono molti titoli dotati di una buona grafica o di un ottimo sonoro, e alcuni che offrono una giocabilità perfetta, ma sembrava che trovare tutte insieme queste caratteristiche fosse un'utopia. Poi è arrivato *Starglider 2*.

Il gioco narra gli avvenimenti successivi alla sconfitta delle truppe Egron, che concludeva la vicenda di *Starglider 1*. La simpatica novella che si trova nella confezione descrive i festeggiamenti e gli onori tributati agli eroici piloti dell'AGAV (l'astronave di *Starglider*), poi i diversi sviluppi delle loro vite, gli avvenimenti politici del pianeta Novenia e molte altre cose, ma per quanto lo si possa apprezzare, il racconto rimane sempre uno dei gadget presenti nella confezione. Tra i gadget, comunque non troppo numerosi, segnaliamo anche il poster e la cassetta delle musiche.

Ma quello che è veramente interessante è il gioco in sé. Dopo il caricamento, che non presenta immagini o musiche che facciano presagire un livello tecnico fuori del comune, ci si trova alla guida di un nuovo e flessibilissimo prototipo spaziale già in volo sulla

superficie del pianeta madre.

La visuale principale è incorniciata dalla strumentazione di bordo, che si avvale di spettacolari strumenti olografici per comunicare le condizioni delle varie parti del veicolo: questa coloratissima cornice racchiude lo schermo di gioco, che è il vero punto di forza del programma. La grafica è stata realizzata in 3D, apparentemente senza limitazioni nell'uso del colore e tantomeno nella velocità di

aggiornamento dello schermo, che si traduce ai fini pratici in un'azione rapidissima e molto realistica.

Ogni elemento del gioco scorre sullo schermo con una fluidità che fa pensare alle animazioni del film *Tron*, che però sono state realizzate con computer grafici dotati di una memoria dell'ordine dei gigabyte!

Il programma prevede un'opzione che permette di scambiare istantaneamente la visione in soggettiva dell'ambiente esterno con una rappresentazione in cui sia presente anche il nostro veicolo, visto sotto qualsiasi angolazione si preferisca. Questa notevole possibilità (niente a che vedere con le relativamente lente "visioni esterne" della Sublogic!) rende possibili soluzioni particolarmente spettacolari, come nel caso in cui, usciti dall'atmosfera, si vede cambiare l'ombreggiatura della nostra astronave a seconda della sua posizione rispetto ai numerosi soli. In casi come questo è comprensibile che il giocatore (e il recensore) non abbia fretta d'iniziare il gioco, e preferisca centellinare tutte le invenzioni dei programmatori.

Ogni elemento grafico, che può anche essere esaminato con calma tramite un'apposita opzione del menu principale, è animato con un realismo incredibile: le navicelle Egron sbattono letteralmente le ali meccaniche durante i loro voli, e la grazia con cui "nuotano" le incredibili balene spaziali di Vista è impossibile da descrivere.

A gioco iniziato, purtroppo, non si ha più il tempo di ammirare la grafica: l'astronave si trova subito nel bel mezzo di un attacco Egron, che si traduce in decine e decine di veicoli e installazioni fisse fermamente decise a distruggerci, che vanno immediatamente affrontate a suon di laser.

Questa è l'unica arma a nostra disposizione all'inizio del gioco, e una rapida esplorazione dell'area d'attacco (un bailamme di esplosioni e crepitii elettrostatici stereofonici) ci convince che è assolutamente necessario trovare mezzi offensivi più affidabili. Ecco che spunta l'elemento strategico del gioco, che ci conduce alla ricerca di sistemi bellici sempre più avanzati. La ricerca, che ci porterà a viaggiare su decine di pianeti e di satelliti, culmina con l'assemblaggio della temibile Bomba Neutronica con cui dovremmo riuscire a eliminare la gigantesca stazione da battaglia Egron che incombe sul pianeta Novenia.

La colossale astronave nemica si trova già in una fase avanzata di costruzione quando il gioco inizia, e se l'equipaggio dell'*Icarus* – il nostro meraviglioso prototipo – non riuscirà a distruggerla prima del completamento, Novenia sarà certamente cancellata da tutte le carte stellari. Una corsa contro il tempo, che ricorda lo spirito di quella favola fantastica che fu *Guerre stellari*.

In ogni fase del programma viene richiesta al giocatore una precisione assoluta nel controllo dell'*Icarus*, indispensabile durante la navigazione nel sistema di tunnel sotterranei presenti su quasi tutti i pianeti, durante le manovre di agganciamento delle armi ausiliarie e soprattutto durante i rifornimenti. Questi avvengono in diversi modi, tutti però con voli a distanza ravvicinata presso potenti fonti di energia: inutile dire che volare troppo vicino a un sole o a un vulcano in eruzione può avere conseguenze spiacevoli...

Starglider 2 non è soltanto un degno seguito del suo eccezionale predecessore: la sua grafica di altissimo livello e la sua grandissima giocabilità permettono di segnalarlo come un modello da seguire e da imitare per qualsiasi gioco spaziale... e non stiamo pensando soltanto al software per l'Amiga.

Qui ci sono anni luce di vantaggio da colmare per tutti: l'impatto audiovisivo è incredibile, e dovrebbe mettere a tacere una volta per tutte i detrattori del computer a 16 bit di mamma Commodore. Insieme con il recente *Carrier Command* e la serie *Ultima*, forma una terna di programmi di gioco che ogni possessore dell'Amiga dovrebbe tenere sempre a fianco del proprio computer.

F.R.

RED STORM RISING

Computer: C-64/128
Supporto: Disco
Prezzo: L. 49.000
Produzione: MicroProse
Disponibile presso: Leader (Via Mazzini 15, 25020 Casciago – 0332/212255)

Ricordate i primi simulatori per home computer? Grezzi, senza un minimo di realismo, i primi veicoli digitali (principalmente aeroplani, ma poi anche elicotteri, navi...) non erano che pallide parodie delle loro controparti reali.

Ora provate a pensare a un simulatore militare tanto realistico da far rischiare al suo autore un processo davanti a una corte marziale per aver diffuso segreti di Stato. Fatto? Adesso pensate che questo simulatore funzioni senza problemi con un computer di piccole dimensioni: se ancora non siete sbottati in frasi come «Impossibile!» o «Figuriamoci!» state pensando a *Red Storm Rising*.

L'ultimo prodotto della MicroProse gira infatti sul modesto C-64, che per l'occasione si trasforma in una copia perfetta di un sommergibile americano. Il realismo del gioco sfiora l'incredibile, e la notizia dell'incidente capitato all'autore del romanzo, Tom Clancy, (fortunatamente risoltosi senza conseguenze) non sorprende più di tanto chi abbia provato il programma almeno una volta.

La storia di *Red Storm Rising* è tratta dall'omonimo romanzo di Tom Clancy, campione di vendita negli Stati Uniti. Il giocatore si trova nei panni di un comandante americano alla guida di un sommergibile che naviga nelle fredde acque del Nord, pullulanti di vascelli russi con intenzioni ostili. Una volta caricato, il programma richiede d'impostare una data riguardante l'imminente azione: se nel 1984 la supremazia in campo sottomarino è totalmente a stelle e strisce, nel 1992 il Patto di Varsavia sarà in possesso di tecnologie tali da tener tranquillamente testa a qualsiasi mezzo di profondità statunitense.

Dopo questa importante scelta, che si ripercuote quindi sulle armi a disposizione delle due fazioni, viene il momento di definire il livello di abilità: le possibilità vanno da un innocuo simulatore in cui i colpi dei nemici non possono danneggiarci, fino a un livello in cui i comandanti sovietici sono pressoché infallibili. Tramite questo menu, *Red Storm Rising* può essere usato come un videogame particolarmente evoluto o come un vero e proprio corso di strategia bellica: sta al giocatore trovare la combinazione di difficoltà che garantisce il massimo divertimento.

L'ultima scelta da fare riguarda il modello di sommergibile. I comandi e la rappresentazione sullo schermo rimangono costanti, ma le varie classi d'imbarcazioni hanno diversi valori di manovrabilità, potenza e silenziosità. Ancora un momento per trovare sul voluminoso manuale il codice d'accesso al gioco (indicato dalla sagoma di un vascello), e finalmente si salpa verso la prima missione.

Potremmo cominciare con un semplice giro di ricognizione, o anche con una missione d'incursione e distruzione in piena regola, ma la situazione tipica di *Red Storm Rising* vede il giocatore solo fra i ghiacci a confronto con i mezzi in uscita da una base semisegreta, nel momento dello scoppio della Terza guerra mondiale.

Mentre il nostro mezzo si allontana dalla base, si possono ammirare numerose schermate di discreta qualità che ci informano sulla situazione dei rapporti Est-Ovest, che sono quasi sempre catastrofici. Questo è un aspetto decisamente negativo del gioco: sin dal momento in cui il programma viene caricato, infatti, la situazione viene presentata secondo un'ottica quasi maccartiana di netta contrapposizione tra i blocchi. I russi sono i cattivi, sono dappertutto e vogliono invadere il giardino di casa nostra; se non stiamo più che attenti prenderanno i nostri bambini ancora in fasce e li mangeranno vivi... In un momento storico come questo, con i nuovi rapporti e le coraggio-

se aperture volute e costruite da Gorbaciov, prodotti come questo andrebbero come minimo rivisti.

Tornando al nostro simulatore, quando anche l'ultima schermata è stata presentata vediamo finalmente l'interno del nostro sommergibile. La situazione è ben diversa da quella dei vecchi *Silent Service* o *Up Periscope!* della Epyx, dove il giocatore si trovava in piena Seconda guerra mondiale; qui il mezzo che guidiamo rappresenta una delle armi tecnologicamente più avanzate dei giorni nostri. I display digitali si illuminano in continuazione mostrando valori sempre diversi, e fra i comandi ce ne sono tre che riguardano il funzionamento del potente computer di bordo. Tutta la battaglia si gioca sul corretto impiego delle tecnologie a nostra disposizione.

La parte principale dello schermo consiste di un display molto simile agli schermi radar dei controllori di volo: ogni vascello è identificato da un simbolo colorato con relativa sigla, in movimento su uno schermo retinato. Questa visuale può essere sostituita da un'immagine proveniente dal periscopio (elettronico anche quello!) o da uno schermo di rapporto proveniente dalle varie sezioni del nostro vascello. Mediante i tasti funzione possiamo richiamare infatti i dati relativi al controllo delle armi, la visione dal periscopio o un rapporto sulle condizioni del mare e degli altri vascelli, di cui possiamo anche ottenere tutti i dati tecnici. Per orientarsi con qualche possibilità di successo in questa moltitudine di comandi, nella confezione viene fornita una mascherina da sovrapporre alla tastiera, senza la quale il gioco diventa praticamente impossibile. Per non parlare del manuale di 108 pagine, fondamentale per riuscire a controllare correttamente il sommergibile. In pratica sono due contromisure particolarmente efficaci contro la pirateria: quanti di voi sono disposti a spendere più soldi per le fotocopie del manuale che non per il gioco originale?

Utilizzando i 46 comandi a nostra disposizione, possiamo guidare il sommergibile ovunque, controllando contemporaneamente i sette tipi di armi offensive e difensive di cui siamo dotati. Fortunatamente queste impressionanti cifre si risolvono, una volta di fronte al programma, in qualcosa di relativamente semplice da controllare.

Ogni volta che lanciamo un missile, o un qualsiasi mezzo viene colpito, una schermata animata ci permette di ammirare "dal vivo" i risultati della nostra azione. Il gioco, basato più sulla tattica che sullo scontro vero e proprio, si trasforma presto in una vera e propria partita di nascondino in cui giocano un ruolo importantissimo le temperature alle varie profondità marine, e l'uso corretto dei mezzi d'identificazione elettronica in dotazione al sommergibile. Impegnato nel coordinamento delle varie sezioni dell'equipaggio, il giocatore scopre ben presto quanto può essere frenetica la vita di un comandante a bordo di un sottomarino. Per fortuna è possibile chiedere aiuto al computer di bordo per avere un consiglio sulla condotta da tenere.

Le battaglie possono durare da pochi minuti a intere ore, che vengono scandite come nella realtà dal rumore ritmico dei sonar di bordo. Il realismo della simulazione è tale che si arriva al punto di sudar freddo per le sorti del proprio mezzo, facendosi nascere terribili scrupoli nelle (rare, per fortuna!) occasioni in cui diventa inevitabile utilizzare le armi nucleari in dotazione.

Red Storm Rising è molto più di un gioco: ogni suo elemento è stato pensato come un vero simulatore militare, ed è in quest'ottica che va considerato. Il manuale, di cui la Leader ha curato una versione ridotta in italiano, è un vero e proprio trattato di strategia. Gli unici appunti da fare sono quelli relativi all'impostazione ultrapatriottica del programma: a questi livelli di realismo non si può prendere tanto alla leggera la trama del gioco. Se si eccettua questa pecca, a *Red Storm Rising* non può andare nessun voto inferiore al massimo.

F.R.

FUSION

Computer: Amiga
Supporto: Disco
Prezzo: n.c.
Produzione: Bullfrog (Electronic Arts)
Distribuzione: Electronic Arts (11/49 Station Road, Langley, Berks SL3 8YN, England)

Finalmente un gioco d'azione degno di questo nome. L'Amiga, come sappiamo, è un computer dalle capacità grafiche veramente notevoli, ed è riuscito a imporsi anche in fasce di mercato medio-basse dato il suo prezzo relativamente accessibile (almeno rispetto a macchine capaci di prestazioni dello stesso livello). Moltissimi però l'hanno acquistato soltanto per utilizzarlo come un sofisticatissimo strumento di gioco, e con questo presupposto è naturale che parte degli sforzi produttivi si siano orientati verso la realizzazione di giochi ad alto livello.

Questo tipo di produzione, però, ha un grave limite dovuto proprio – per ironia della sorte – alla qualità della macchina. Infatti, i programmatori sono indotti dalle caratteristiche dell'Amiga a curare le proprie realizzazioni soprattutto dal punto di vista grafico, e a presentare giochi dalle animazioni stupefacenti... ma spesso carenti dal punto di vista dell'azione e della giocabilità.

Da questa tradizione negativa prende decisamente le distanze *Fusion*, una delle ultime realizzazioni messe in commercio con la garanzia del nome Electronic Arts. Si tratta di uno shoot'em up di tipo moderno, nel quale alla rapidità di gioco si unisce una buona varietà di azioni e di situazioni, che creano una trama fitta e avvincente in un crescendo davvero divertente.

Lo scopo del gioco è quello di trovare e raccogliere le varie sezioni di una bomba, sparse nei diversi settori di cui si compone il gioco. Se si riesce a riunire tutte le componenti nel settore di partenza, ci si può considerare vincitori.

I mezzi a disposizione per compiere l'impresa sono un assault crawler, cioè un piccolo mezzo semovente con limitate possibilità di movimento all'interno del settore (non può superare muri, salite, discese...), e un'astronave madre alla quale si può agganciare il mezzo semovente, che invece può spaziare in lungo e in largo nel settore producendo un notevole volume di fuoco. Per passare da un settore all'altro, oppure per raggiungere speciali bonus o componenti della bomba, è spesso necessario aprire alcune "porte", utilizzando i cosiddetti "interruttori". Si tratta di blocchetti quadrati dalla colorazione particolare che possono essere catturati solo dall'assault crawler.

Poiché quest'ultimo si può sganciare dalla nave madre soltanto sopra spazi sgombri e piuttosti ampi, la tecnica per catturare gli interruttori, è quella di ripulire con la nave madre (più mobile e meno soggetta ai colpi) il percorso che porta dallo spiazzo all'interruttore, e poi inviare il piccolo semovente a catturarlo.

Proprio in questo sta una delle migliori caratteristiche del gioco, la possibilità cioè di manovrare due sprite di forma e caratteristiche diverse contro un ben assortito gruppo di nemici, e quella di mettere alla prova memoria e abilità nella ricerca dei percorsi e degli interruttori appropriati. I nemici da affrontare sono molto diversi tra loro: spaziano dalle più innocue postazioni fisse, ai temibili UHO, veloci e con ampia libertà di movimento; e ogni nemico impone un diverso modo di combattere.

Dal punto di vista visivo, il gioco è ben congegnato. Non offre certo immagini di tipo "cinematografico", ma la grafica è adatta al tipo di gioco ed è molto chiara e pulita sia nel disegno sia nell'animazione, riuscendo così a mettere in rilievo le migliori caratteristiche dell'Amiga – ovvero nitidezza dell'immagine e fluidità di movimento – che spesso vengono trascurate a favore di schermate complesse ma quasi statiche.

In definitiva si tratta di un gioco ben riuscito e divertente. Chi ama il contatto fisico con il joystick e vorrebbe muoverlo in continuazione non rimarrà certo deluso da *Fusion*.

N.F.R.

VIRUS

Computer: Amiga
Supporto: Disco
Prezzo: L. 29.000
Produzione: Firebird
Disponibile presso: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Un programma di buon effetto. A prima vista addirittura affascinante, con la sua totale tridimensionalità: ogni oggetto può essere visto da qualsiasi angolazione semplicemente girandoci intorno con la navicella controllata dal giocatore. La fluidità di movimento messa in luce dal programma, e valorizzata dalle eccellenti capacità messe a disposizione dell'Amiga, forma con la tridimensionalità un'accoppiata senza alcun dubbio di grande effetto.

Ma l'impressione iniziale, a parte l'ovvia constatazione di una notevole raffinatezza a livello tecnico, non è sufficiente per assegnare all'ultimo prodotto della Firebird un voto superiore al cinque, e si tratta comunque di un voto d'incoraggiamento. Il principale motivo d'insufficienza sta proprio in quelli che dovrebbero essere i punti di forza di un gioco come questo, ovvero l'azione, lo stimolo, l'imprevisto. Ancora una volta siamo di fronte a un'esercitazione di abilità da parte di bravissimi programmatori... che hanno dimenticato l'obiettivo del lavoro che stavano facendo, e si sono lasciati prendere la mano dal desiderio di sfruttare fino in fondo le possibilità grafiche dell'Amiga (si può comunque fare di più).

La trama è presto detta: il giocatore deve pilotare un Hoverplane, un'aeromobile dell'ultima generazione, con il compito di difendere il territorio dagli attacchi alieni. Lo scopo degli invasori spaziali, invece, è la distruzione della vita sulla Terra. La loro arma è decisamente inconsueta: niente laser, raggi della morte o cannoni fotonici, ma soltanto un'arma biologica, un inarrestabile virus che attacca la flora terrestre.

L'aeronave è equipaggiata con un radar a lungo raggio (in alto a sinistra, sullo schermo), con alcuni cannoni laser e missili a ricerca automatica del bersaglio. Lo scopo del gioco è l'eliminazione nel tempo più breve possibile delle astronavi nemiche che stanno seminando il virus. Una volta polverizzata l'ultima astronave, il giocatore guadagna un bonus e passa allo schermo successivo, pronto ad affrontare difficoltà via via crescenti.

Il tipo di astronavi con cui ci si deve confrontare è piuttosto vario, ma l'azione – come si è già detto – risulta ugualmente monotona. Anche la grafica non è delle migliori, nonostante la tridimensionalità, soprattutto per lo scarso realismo.

Un motivo di pregio, invece, è il sistema di guida dell'aeromobile, che si comanda completamente tramite mouse. Muovendo il mouse si può inclinare la nave longitudinalmente (mouse avanti e indietro) o trasversalmente (mouse a destra e a sinistra) fino a farla ruotare completamente. Premendo il tasto sinistro si aziona il propulsore: la direzione viene impostata regolando le varie inclinazioni con il mouse, e dal momento che la nave risponde molto rapidamente al minimo movimento del mouse, è piuttosto impegnativo anche per il miglior giocatore trovare il giusto assetto.

Sono gli unici momenti emozionanti in un gioco che, a nostro avviso, ha puntato troppo sulla tridimensionalità, senza ricordare che la novità e la fantasia contano sempre più degli "effetti speciali". È una regola che si dovrebbe sempre tenere a mente, se si lavora nel campo dello spettacolo. Cinema e videogiochi sono generi diversissimi, ma hanno lo stesso scopo: intrattenere in modo piacevole e, se possibile, intelligente. Nel caso di *Virus* non ci troviamo dinanzi né al divertimento né a qualcosa d'intelligente. Meglio spendere il proprio tempo con altri programmi, cercando di dimenticare anche il nome di questo prodotto. Non vi viene in mente l'assonanza con le temibili infezioni che corrono di disco in disco?

N.F.R.

ROCKET RANGER

Computer: Amiga
Supporto: Disco
Prezzo: L. 59.000
Produzione: Cinemaware (Mirrorsoft)
Disponibile presso: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Avventura grafica, gioco di strategia o cinema interattivo? Questi gli interrogativi che ci siamo posti di fronte alla nuova opera della Cinemaware, *Rocket Ranger*. Dai tempi di *Defender of the Crown* e di *S.D.I.* stiamo seguendo con interesse il lavoro di questa software house e i suoi progressi in direzione di nuove forme d'intrattenimento interattivo. *King of Chicago* aveva fatto ben sperare, poi con *The Three Stooges* ci era sembrato di notare una battuta d'arresto. In quest'ideale cammino a tappe verso una nuova generazione di computer game interattivi, *Rocket Ranger* rappresenta il superamento delle precedenti incertezze e il raggiungimento di un difficile punto di equilibrio tra fascino della storia, perfezione grafica e giocabilità.

28 dicembre 2040. Questa data è riportata in calce al drammatico documento che dà inizio alla storia. Si tratta dell'appello che un gruppo di scienziati del lontano futuro manda a noi uomini del 1940. Già a una rapida lettura ci rendiamo conto che i nostri incubi più foschi si sono avverati: il nazismo ha vinto la Seconda guerra mondiale e il Terzo Reich stringe in pugno il mondo intero nella morsa della sua dittatura implacabile. Gli scienziati che invocano il nostro aiuto si trovano proprio nello stesso luogo in cui ci troviamo noi: a Fort Dix nello Stato del New Jersey, ma a un'incolmabile distanza di cento anni nel futuro.

Incolmabile? Forse no, perché questo gruppo di nostri bisnipoti sta lavorando a una nuova invenzione... la macchina del tempo. È proprio grazie a questo strumento che ci è giunto il loro appello, accompagnato da tre oggetti avveniristici: l'*Himmelwurfer Rocket Pack*, un ordigno capace di trasportarci nel cielo grazie al suo potentissimo carburante, che i nostri amici del 2040 chiamano "lunarium", una *Schmeisser Radium Pistol*, l'arma che ci accompagnerà nel corso dell'avventura, e un *Telefunken Wrist Monitor*, un vero e proprio computer da polso.

È stata la scoperta del lunarium a spostare a favore delle feroci armate hitleriane i rapporti di forza e a provocare così il tracollo del mondo libero, e tutto sarebbe avvenuto proprio nel 1940, lo stesso anno in cui inizia la nostra avventura.

Non è un caso se il messaggio è giunto sulla nostra scrivania. La figlia di Otto Barnstoff, il più grande scienziato americano, è stata infatti nostra compagna di studi, e Otto Barnstoff in qualche strano modo è collegato al lunarium...

Mettiamo ora da parte il mes-

saggio dei nostri lontani discendenti, il prezioso libretto che accompagna il gioco e la ruota decodificatrice (Secret decoder wheel) inclusa nella confezione e addentriamoci nel gioco.

L'obiettivo che ci è stato assegnato è chiaro: dobbiamo fermare a tutti i costi la terribile macchina da guerra nazista prima che prenda il controllo dell'intero pianeta. Il piano prevede di raccogliere in cinque località segrete sparse per il globo le parti necessarie per la costruzione di un'astronave grazie alla quale potremo raggiungere la Luna, la sede del potere nazista. Ma prima è necessario procurarsi anche un'adeguata quantità di carburante, ovvero il lunarium.

Il luogo di partenza del gioco è Fort Dix. Un menu ci consente di scegliere tra quattro diverse locazioni (e quindi situazioni di gioco): la War room, il Fuel depot, il Rocket lab e infine il Takeoff. Esaminiamole.

War room. Non siamo soli nella nostra lotta contro il potere nazista. Abbiamo ai nostri ordini cinque agenti segreti che opportunamente dislocati nel mondo possono aiutarci nella missione, recuperando per esempio le diverse parti meccaniche dell'astronave che ci porterà sulla Luna. Tramite la Secret decoder wheel possiamo accedere a ben 26 regioni della Terra e un sistema di icone ci metterà al corrente delle informazioni disponibili. Purtroppo possiamo comunicare con i nostri agenti solo da questa stanza, e quindi sarà bene che mettiamo in bilancio molteplici viaggi di ritorno se vogliamo entrare in possesso delle informazioni che a poco a poco si renderanno disponibili.

Rocket lab. Supponiamo di aver avuto successo nella missione precedente e di aver quindi rintracciato le diverse parti dell'astronave: ebbene, questo è il laboratorio dove le metteremo insieme. Sempre qui, durante il lungo lavoro, riceveremo alcune informazioni indispensabili che ancora ci mancano.

Fuel depot. Il lunarium non è un materiale facilmente trattabile. Dobbiamo infatti tornare ogni volta a questo deposito per effettuare le operazioni di carico e scarico del Rocket pack o, in seguito, della Rocket ship.

Takeoff. È la prima delle imprese in cui dobbiamo cimentarci nella nuova veste di eroi del cielo. Dopo diversi insuccessi riusciamo finalmente a prendere il volo e partiamo verso il compimento di una missione vitale per l'umanità.

Agenti segreti, codici di comunicazione, assemblaggio della nave spaziale. Sono questi gli elementi portanti di *Rocket Ranger*. Ma non c'è solo questo: finché noi ci prepariamo a sferrare il nostro attacco, infatti, il nemico nazista non se ne sta certo con le mani in mano. Ci attendono molti impegnativi combattimenti, a terra e nel cielo, e solo un accorto impiego della nostra Radium pistol ci permetterà di affrontarli con successo.

Abbiamo parlato della trama, ne abbiamo messo in luce gli elementi di strategia e gli aspetti tipici degli arcade... ora è finalmente il momento di parlare anche della qualità della grafica. Estremamente curata in ogni particolare, ispirata da una parte ai canoni dell'Art deco, dall'altra ai fumetti degli anni Quaranta e Cinquanta, rappresenta sicuramente un punto di riferimento nel panorama attuale dei giochi per computer.

Il nostro parere, a questo punto, non può che essere estremamente favorevole, anche se avremmo preferito una maggiore presenza dell'animazione. Ci dicono tuttavia che la Cinemaware sta provvedendo anche a questo, però soltanto su CD ROM e solo in via sperimentale!

F.T.

TYPHOON

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 12.000/15.000
Produzione: Imagine (Konami)
Disponibile presso: Leader (Via Mazzini 15, 25020 Casciago – 0332/212255)

Typhoon: chi frequenta le sale giochi avrà sicuramente drizzato le orecchie nel leggere questo titolo. E non sbaglia. Questo programma infatti è la versione per home computer di uno dei giochi a gettone di maggior successo del 1988, premiato con un "Arcade Award" nel mese di settembre.

Quello che ha fatto la fortuna di questo coin-op è sicuramente la sua grafica innovativa e spettacolare, resa possibile dal sistema "bubble memory" sviluppato dalla Konami negli ultimi anni: questa nuova tecnologia, tenendo conto anche del buon livello del sonoro, ha fatto dire a più di un recensore della stampa specializzata inglese che la conversione a otto bit del gioco era un sogno senza speranze.

Ebbene, dopo aver messo alla prova questa versione per il piccolo Commodore 64 per diversi giorni, in ripetute e massacranti sessioni di gioco, siamo veramente lieti di poter dire che si sbagliavano.

Il gioco prende il via senza inutili complicazioni: la Terra viene invasa da truppe aliene Mecanoidi, e nella migliore tradizione degli arcade il giocatore si trova alla guida di un mezzo pesantemente armato con il quale deve infiltrarsi tra le linee nemiche e controbattere l'invasione.

Bisogna notare subito una cosa: nonostante la trama, *Typhoon* non è assolutamente un gioco di ambientazione spaziale, e chi si aspetta battaglie all'ultimo laser contro dischi volanti e astronavi aliene farà meglio a continuare a leggere per scoprire di che cosa si

tratta in realtà.

Il primo livello del gioco ci mette infatti alla guida di un caccia statunitense F-14 terribilmente simile al protagonista del bellissimo *Afterburner* della Sega. Con questo colosso d'acciaio il giocatore si lancia in picchiata attraverso grigi banchi di nuvole verso la superficie del mare: una perfetta sequenza a tre dimensioni che ci costringe a mettere veramente a dura prova le bombe e le mitragliatrici di cui dispone il velivolo, utilizzandole senza risparmio contro formazioni di aerei nemici che spuntano dalle nuvole.

Una volta oltrepassata l'ultima barriera di nubi, il nostro aereo si trova di fronte alla spettacolare scena dell'avvicinamento alla portaerei nemica (per quanto meno entusiasmante di quella messa in mostra dal fratello maggiore nelle sale giochi). La nave appare all'inizio come un piccolo e insignificante gruppetto di pixel grigi, che si trasformano però molto presto (non dimenticate che il nostro jet F-14 sta ancora scendendo in picchiata!) in una gigantesca struttura che occupa quasi l'intera grandezza dello schermo di gioco.

In questa fase scompaiono gli aerei nemici per fare posto a decine e decine – letteralmente! – di missili contraerei lanciati contro di noi dalla nave: per eliminarla, evitando nel frattempo di schiantarci sulla sua superficie, dovremo colpire con una bomba un piccolo bersaglio rosso presente sul ponte.

Con la distruzione della portaerei avversaria si supera questa sezione, e dopo un ulteriore caricamento, il gioco cambia completamente: non solo ci mostra l'azione dall'alto anziché in soggettiva forzata, ma il nostro mezzo aereo si è trasformato in un grosso elicottero trasportato addirittura da un'enorme astronave lucente.

Qui cominciano finalmente a rivelarsi le caratteristiche più importanti di *Typhoon*. La cosa che salta subito all'occhio è l'estrema rapidità dell'azione e il grande numero di sprite in movimento sullo schermo. Per una volta possiamo dire senza paura di essere smentiti che il ritmo "da coin-op" dell'originale è stato mantenuto perfettamente e che anche il più incallito giocatore troverà senza dubbio pane per i suoi denti.

L'elicottero viene presto minacciato pericolosamente da jet, sottomarini, altri elicotteri e navette-pattuglia provenienti da ogni direzione, e ancora una volta la rapidità di fuoco delle bombe e delle mitragliatrici si rivela essenziale per sopravvivere.

Questa sequenza, in cui compaiono anche installazioni terrestri contraeree, ricorda molto da vicino il vecchio *Xevious*, eccezionale soprattutto nella sua versione a gettoni. I proiettili nemici sembrano circondare senza speranza l'elicottero del giocatore, e l'azione frenetica viene spezzata solo dalla detonazione della megabomba di cui è dotato l'elicottero, che offre al gentile pubblico una delle esplosioni più apocalittiche di tutta la storia dei videogame. Incidentalmente elimina anche tutti i nemici presenti sullo schermo.

Nel procedere verso la fine del livello, al giocatore viene offerta la possibilità d'incrementare la potenza delle proprie armi raccogliendo alcuni moduli lasciati cadere da particolari nemici: a seconda della rapidità dell'operazione, questi doneranno all'elicottero l'interessante possibilità di sparare con armi di diversa potenza, che vanno dalla scarica di missili (Vulcan fire) a un semplice fuoco rapido delle mitragliatrici.

Alla fine del livello appaiono i primi alieni "convenzionali" (finalmente). E non sono troppo bellicosi: si presentano infatti sotto forma di globi rotanti e barriere elettroniche che vanno eliminate a colpi di bomba, mettendo in campo solo un po' di precisione.

Poi un nuovo caricamento ci porta in una serie di apocalittici scenari popolati da nemici sempre più veloci e intelligenti, e infine si torna alla guida dell'originario F-14 per lanciarsi in una corsa rasoterra verso una gigantesca torre armata.

Ma a questo punto siamo costretti a concludere la nostra recensione perché, pur vantando un invidiabile passato di campioni di videogame, purtroppo non siamo riusciti a progredire oltre nel breve tempo che avevamo a disposizione.

Come avrete già capito dalla descrizione dei primi due livelli, *Typhoon* è un gioco di azione pura, che riesce a portare la produzione di adrenalina a livelli veramente stratosferici, e ricrea perfettamente la frenetica atmosfera di un gioco arcade a gettoni. Nonostante la grafica sia sempre di livello accettabile, sono soprattutto gli effetti di animazione e di tridimensionalità a fare la parte del leone.

Il sonoro di *Typhoon* è senz'altro degno di particolare nota. Oltre agli ottimi effetti, comprende una serie di musiche – diverse per ogni livello – che contribuiscono alla coinvolgente tensione che caratterizza questo gioco anche nella versione per home computer. L'impressione globale è quella di un prodotto di altissima qualità che dovrebbe tenere impegnato qualsiasi giocatore per diverso tempo.

Un'ultima caratteristica interessante, che abbiamo accolto con molto piacere e che costituisce un aiuto notevole per non perdere troppo tempo nella fase di familiarizzazione con il gioco, è la possibilità di ricominciare la partita dall'ultimo progresso raggiunto (ma solo nei primi due livelli) e il gran numero di "vite" che viene fornito per ogni partita, che arriva a sei contro le tre canoniche.

F.R.

SOFTWARE HELPLINE

Sherlock Holmes: la soluzione

Nota: Gli orari indicati hanno valore soltanto se la soluzione viene seguita passo per passo, ma sono comunque sufficienti a dare un'idea di quando devono essere compiute determinate azioni. La raccolta delle alghe, ad esempio, è un'azione subordinata ai cicli delle maree, che possono essere controllati sull'apposita tabella oraria contenuta nella confezione del gioco.

221 B Baker St.
Knock on door. Up. N. Tell Holmes about visitor. Get pipe, tobacco and newspaper. Put tobacco into pipe. Wait. Read verse. Wait. W. Get ampoule, lamp and glass. Examine ampoule. E. S. D. N. Get matchbook and read it. S. Open door. E. Light lamp. S (tre volte). SE (due volte). Take off hat and get stethoscope. Put ampoule in hat. Wear hat. E (due volte). Read sign. Ask Holmes about ampoule. Wait until 7:00. Go east.

Nave and Westminster Abbey
S. SE. Get pacquet, crayon. Open pacquet and get brown, yellow and orange. NW. Open door. S. W. Read sign. E. N. N. Examine tomb. Put brown paper on tomb and rub it with crayon. Get paper. E. N. E. Examine tomb. Put orange paper on tomb and rub it with crayon. Get paper. S. W. Examine tombs. Put yellow paper on Henry's tomb. Rub it with crayon. Get paper. E. N. W. N. Look. Heat orange paper with candles. Read back of orange paper. (Ripetete queste due ultime operazioni anche con le carte marrone e gialla). S (due volte). W (due volte). NE. N. E. D. Look into boat. Get oar. U. Turn off lamp. W. Blow whistle (due volte). Get in cab. The Embankment. Get out.

The Embankment
Get into boat. Put oar in oarlock. Weigh anchor. Launch. Row east (due volte). Drop anchor. Examine bridge. Examine moss. Wait until 8:40 (o la successiva alta marea, se avete perso tempo precedentemente). Get moss. Weigh anchor. Row west (due volte). Land. Get out. Examine opal. Examine opal through magnifying glass. Blow whistle (due volte). Get in cab. Birdcage Walk. Get out.

Birdcage Walk
Haggle with vendor (due volte). Buy telescope. E. N. N. Examine Nelson through telescope. NE. N.

Convent Garden
Wear stethoscope. Listen to girl. Open bag. Open blue bottle. Drop crayon, pacquet and all paper. Get cotton and newspaper. (Se il battito cardiaco della ragazza è troppo rapido, dovete darle la pillola arancio; in ogni altro caso, aprite la bottiglia marrone e datele la pillola gialla). Take off stethoscope. N. E. S. W.

Sherman's Shop
Ask Sherman about pigeon. Ask Sherman for pigeon. E. Blow whistle (due volte). Get in cab. Trafalgar Square. Driver, wait here. Get out. Show ruby to pigeon. Pigeon, get ruby. Let go of pigeon. Get in cab. Pinchin Lane. Driver, wait here. Get out. W. Ask Sherman about pigeon. Examine ruby through glass. E. Get in cab. Marylebone Road. Get out.

Marylebone Rd.
N. Ask guard about cigarette. Open matchbook. Light match. Light pipe. Drop matchbook and lamp. N. Ask Holmes about ash. W. Get torch. Light newspaper with pipe. Light torch with newspaper. Get head. Melt head with torch. Get gem. Examine gem through glass. E. S. Get lamp. Blow whistle (due volte). Get in cab. Parliament Square. Get out.

Parliament Sq.
SE. Up. Put cotton in ears. Wait unil 12:00 (o sino alla successiva ora). Get sapphire. Wait. Get sapphire. D. Remove cotton. Examine sapphire through glass. NW. N. N. NE. N (due volte). E (due volte).

Threadneedle St.
Examine urchin. Ask Holmes about guard. Wiggins, get keys. Give shilling to Wiggins. Wiggins, get keys. N. Give all gems to guard. N. Wear stethoscope. Listen to door. Turn dial right (due volte). Turn dial left. Turn dial right (due volte). W. Take off stethoscope. Unlock box 600 with key. Get topaz. Examine topaz through glass. E. S. Blow whistle (due volte). Get in cab. Trafalgar Square.

Trafalgar Sq.
Driver, wait here. Get out. W. Ask butler for Mycroft. Give ring to butler. (Tenete a mente la parola d'ordine). E. Get in cab. Tower of London. Get out.

Tower of London
E. E. (Dite la parola d'ordine di Mycroft). N (due volte). SE. U. Get weapon. D. NW. NE. Hit bung with mace. Look in keg. Wiggins, get garnet. Examine garnet through glass. SW. E. D. Wear armor. U. W. S (tre volte). Get paddle. Pull chain. Take off armor. S. Get into boat. Weigh anchor. Launch. Paddle west (tre volte). Land. Get out. E (due volte). D. Wait until 2:00 AM Monday. W.

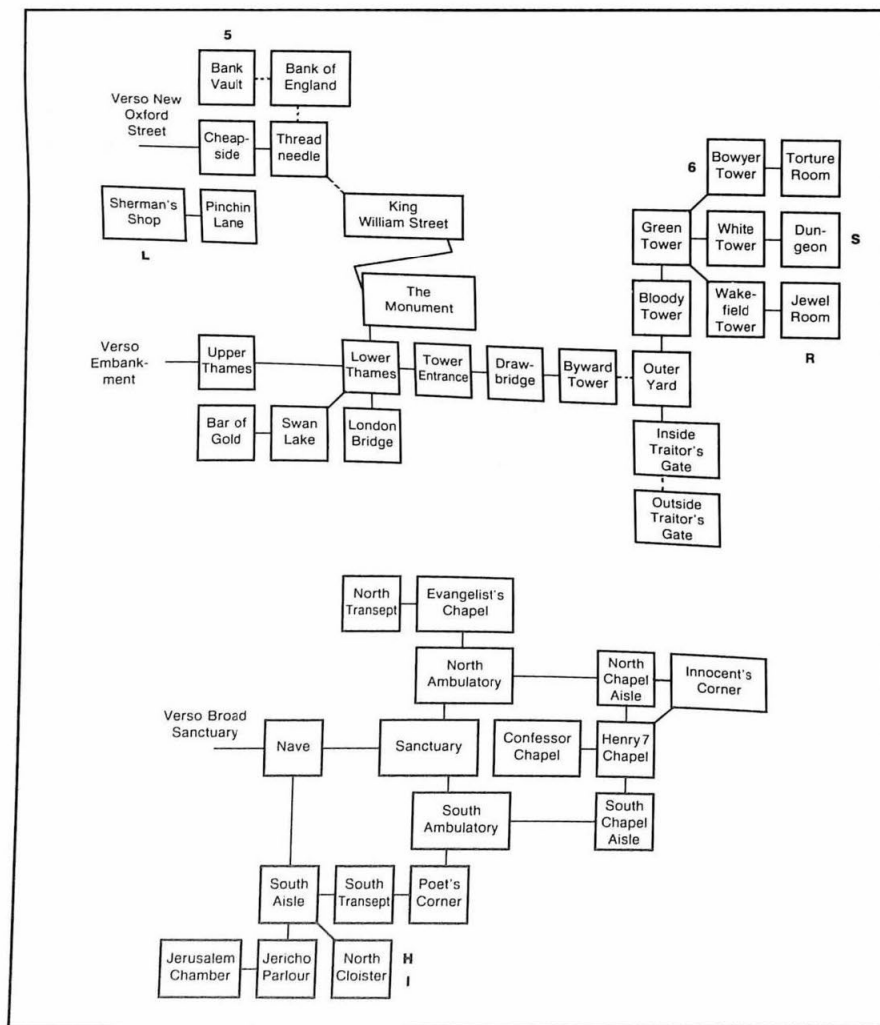
Bar of Gold
Ask for Akbar. Swordfish. Give garnet to Akbar. Ask Moriarty about jewels. Take off hat. Get ampoule. Hold breath. Break ampoule. Untie Holmes. Tie up Moriarty and Akbar. Get jewels and whistle. Get key. Unlock door with key. Open door. Out. Blow whistle (due volte). Get in cab. Queen's Gardens. Get out. Give jewels to guard.

Legenda degli oggetti sulla mappa
A: pipe
B: tobacco
C: newspaper
D: ampoule
E: lamp
F: glass
G: matchbook
H: pacquet
I: crayon
J: oar
K: telescope
L: pigeon
M: ash
N: torch
O: axe
P: head
Q: password
R: mace
S: armor
1-6: jewels

SoftMail

Vendita per corrispondenza di programmi originali ed accessori per computers.

VIA NAPOLEONA 16 - 22100 COMO - TEL. (031) 30.01.74

® SoftMail è un marchio registrato da Lago snc

Presentiamo in questa pagina alcune tra le ultime novità disponibili dal catalogo SoftMail. Ecco alcune informazioni utili per utilizzare il servizio SoftMail: è possibile effettuare ordini telefonicamente SOLO se è già stata effettuata una spedizione a proprio nome ed è stata regolarmente ritirata. Dal secondo in poi accettiamo ordini telefonici. Chi desidera notizie sulla disponibilità ed i prezzi dei prodotti che non compaiono in questa lista può chiamare lo (031)30.01.74 dalle 14:30 alle 18:00. SoftMail può organizzare la consegna anche tramite corriere: interpellateci per maggiori informazioni. Oltre alle ultime novità qui esposte, SoftMail offre l'intero catalogo delle seguenti case: Aegis, Cinemaware, EA, Microprose, Rainbird, SSI, SSG, Sublogic.

Dal 15 novembre fino al 24 dicembre in ogni pacco c'è un *bellissimo regalo:* Buon Natale da SoftMail!

ACCESSORI

Copritastiera A500	25.000
Final cartdrige III	110.000
Flicker master	29.000
Joy. Slik stick	16.500
Joy. SpeedKing	29.000
Joy. SpeedKing Autof.	33.000
Joy. Tac 2	29.000
Joy. Tac 5	39.000
MouseMat	22.500
MouseTop	20.000
MouseHouse	20.000
MouseHolder	12.500
Portadischi 3" (30)	34.000
Portadischi 5" (40)	37.000
SlimLine (tastiera 64)	49.000
Voice messenger C64	60.000

LIBRI/HINTS & TIPS

Alternate city clue specificare 8/16 bit	18.000
Bard's tale I clue	22.500
Bard's tale II clue	25.000
Bard's tale III clue	25.000
Dungeon master clue	25.000
IHT: tutti i titoli...	telef.
Might & Magic clue	25.000
Quest for clues	39.000
Sentinel world clue	25.000
Starflight clue	25.000
Wasteland clue	16.500

AMIGA

Aegis draw 2000	telef.
Aegis draw: Art/CAD pack	telef.
Aegis images	69.000
Aegis impact	125.000
Aegis sonix	110.000
Aegis videotitiler 1.1	185.000
Alternate reality: the city	69.000
Animation multiplane	125.000
Armageddon man	49.000
Audiomaster	85.000
Autoduel	59.000
Barbarian II	telef.
Battlechess	49.000
Black lamp	39.000
Blitzkrieg at Ardennes	59.000
Butcher	59.000
Captain blood	49.000
Carrier command	49.000
Cell animator	telef.
Centerfold squares	59.000
Comic setter	139.000
Comic setter art disks	telef.
Corruption	49.000
Crazy cars	29.000
Def con 5	59.000
Defender of the crown	59.000
Deja vu II	telef.
Deluxe print II	99.000
Deluxe productions	299.000
DigiPaint	99.000
DigiView GOLD	telef.
Director	99.000
Disk drive esterno	299.000
Dos2Dos	75.000
Dragon's lair (1 MB, 6 disk)	telef.
Dragonslayer	telef.
Driller TUTTO italiano	59.000
Dungeon master	telef.
Elite	45.000
Empire	49.000
Express paint 3.0	telef.
Fantavision	125.000
Fire & forget	39.000
Fish	45.000
Flight simulator II	99.000
Scenery disks	telef.
Forms in flight	125.000
Galileo 2.0	99.000
Garfield	49.000
Grabbit	49.000
Hellfire attack	39.000
Hostages	telef.
Impossible mission II	25.000
International soccer	39.000
Italy 90 soccer	39.000
Interchange vari...	telef.
Jet & Japan bundle	110.000
Jinxter	45.000
King of Chicago	49.000
Lattice 5.0	telef.
Leaderboard birdie	29.000
Legend of the sword	45.000
Life cycles vol.1	39.000
Light!Camera!Action!	99.000
Lords of the Rising Sun	telef.
Major motion	39.000
Maxiplan plus	299.000
Menace	39.000
Mission: con-bat	telef.
Modeler 3D	129.000
Mortville manor	59.000
Nigel Mansell grand prix	49.000
Obliterator	45.000
Offshore warrior	39.000
Outrun	25.000
Olympic challenge	25.000
Operation cleanstreet	telef.
P.O.W.	59.000
PacMania	25.000
Page flipper Plus	79.000
President is missing	telef.
Ports of call	65.000
ProSound designer	79.000
ProSound w/hardware	199.000
Prowrite 2.0	179.000
Publisher plus	125.000
Reach for the stars	59.000
Return to Genesis	39.000
Rocket Ranger	59.000
SEUCK	telef.
S.D.I.	59.000
Sculpt/Animate 4D	telef.
Sentinel	49.000
Shadowgate	49.000
Sinbad	59.000
Skyfox II	59.000
Speedball	49.000
Star Goose	39.000
Star ray	49.000
Starfleet	59.000
Starglider II	49.000
Strike force harrier	49.000
The bard's tale II	49.000
The works!	249.000
Thexder	59.000
Three Stooges	59.000
Time & magic	45.000
Times of Lore	telef.
Tracker	49.000
TV Sports: Football	telef.
TV*Show	125.000
TV*Text	125.000
Ultima IV	49.000
Universal military sim.	45.000
Data disks 1 e 2	telef.
Videoscape 3D 2.0	250.000
Designs disks	telef.
VIP professional	199.000
Virus	29.000
Virus infection protection	89.000
WB Extras	69.000
Whirligig	39.000
Who framed Roger Rabbit	telef.
World Class Leaderboard	telef.
Write & file	125.000
Zing keys	25.000
Zoetrope	telef.
Zoom	45.000
3 Demon	145.000
4x4 Off road racing	25.000

COMMODORE 64 CASS.

Around the world in 80	25.000
Barbarian II	18.000
Dark side	18.000
Driller	25.000
Flight simulator II	39.000
Game over I+II	18.000
Gold Silver Bronze	18.000
Hawkeye	18.000
Impossible mission II	12.000
Last Ninja II	25.000
MicroSoccer	telef.
Red storm rising	39.000
Shoot'em up constr. kit	25.000

COMMODORE 64 DISCO

ADD: Heroes of the lance	telef.
ADD: Pool of radiance	59.000
American civil war III	49.000
Barbarian II	25.000
Captain blood	29.000
Carrier force	55.000
Caveman ugh olympic	telef.
Chessmaster 2100	69.000
Deathlord	35.000
Driller	29.000
Echelon (con LipStick)	59.000
Eternal dagger	35.000
Faery Tale adventure	telef.
Fast break	29.000
Game over I+II	25.000
GEOS 2.0	95.000
GeoPublisher	100.000
Grand Prix Circuit	telef.
Gulf strike	49.000
Home video producer	69.000
Infocomics vari...	telef.
Mars saga	35.000
MicroSoccer	telef.
Neuromancer	39.000
Panzer strike	45.000
Red storm rising	49.000
Rocket Ranger	telef.
Shoot'em up constr. kit	35.000
Stealth mission	75.000
The bard's tale III	35.000
Ultima V	49.000
Wasteland	39.000
Who framed Roger Rabbit	telef.
Zak Mc Kracken	39.000

COMMODORE 128 DISCO (128K, 80 COL.)

"C" Language	99.000
Basic 8.0	75.000
GEOS 128	telef.
Mach 128	125.000
Superpack II	179.000

OFFERTE SPECIALI NATALIZIE

(FINO AD ESAURIMENTO)

FIREBIRD SPECIAL "NATALE 1988"		Valore	A sole
AMIGA	Bubble Bobble+Whirligig+Enlightenment	107.000	75.000
C64 cass.	Savage+Intensity+Sentinel	48.000	33.000
C64 disco	Savage+Intensity+Sentinel	65.000	39.000
AEGIS ENTERTAINMENT			
AMIGA	Arazok's Tomb+Ports of Call	120.000	89.000

DEFENDER OF THE CROWN C64: cass. 12.000 disco 15.000

... FAI UN BEL REGALO AL TUO COMPUTER!

BUONO D'ORDINE da inviare a: LAGO DIVISIONE SOFTMAIL, VIA NAPOLEONA 16, 22100 COMO, FAX (031) 30.02.14, TEL (031) 30.01.74

Desidero ricevere i seguenti articoli:

- ☐ Addebitate l'importo sulla mia CARTASI/MASTERCARD/VISA nr. ______________ scadenza ________
- ☐ Pagherò al postino in contrassegno

Titolo del programma	Computer	Cassetta/disco/accessorio	Prezzo

Contributo spese di spedizione Lit. 5.000

TOTALE LIT. ________

ORDINE MINIMO LIT. 20.000 (SPESE ESCLUSE) G

Cognome e nome ______________________

Indirizzo ______________________ Nr. ________

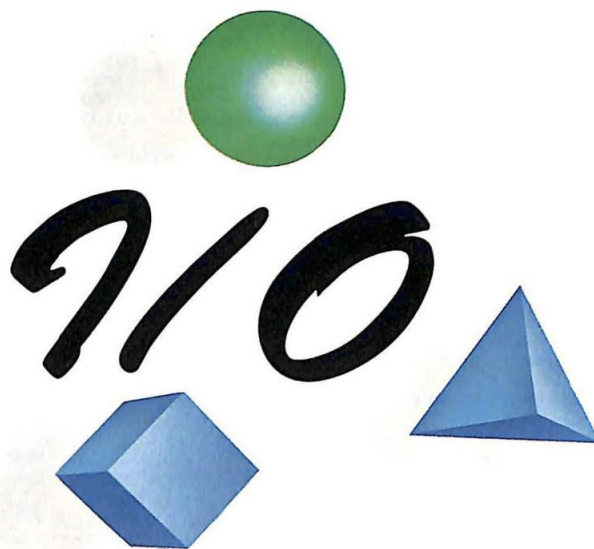
CAP ______ Città ______________ Prov. ________ Telefono ________

FIRMA (SE MINORENNE QUELLA DI UN GENITORE)

VERRANNO EVASI SOLO GLI ORDINI FIRMATI ______________________

031/30.01.74

I N P U T / O U T P U T

INPUT/OUTPUT SVELA I SEGRETI DEL VOSTRO COMPUTER

C-64/128 in modo 64

***186 Stop con i file difettosi -** I programmi che salvano dati sul disco generalmente non effettuano alcun controllo sulla quantità di spazio libero effettivamente disponibile. Il risultato è che se il numero di blocchi liberi è insufficiente, il salvataggio dei dati non ha successo, e sulla directory del disco compare un file difettoso evidenziato da un inquietante asterisco. È noto a tutti che la presenza di questi file su un disco rappresenta un problema, in quanto cancellandoli o impartendo il comando Validate c'è sempre il rischio (benché remoto) di cancellare anche dati appartenenti ad altri file.

La routine Basic che propongo, opportunamente inserita in un programma prima di ogni accesso al disco in scrittura, effettua una verifica dello spazio ancora disponibile su disco, e se non è sufficiente avvisa l'utente. Perché funzioni, dev'essere chiamata impostando nella variabile LL il numero di blocchi del disco che si prevede saranno necessari per salvare i dati correttamente.

```
100 LL=30:REM BLOCCHI DA SALVARE
110 OPEN 1,8,0,"$0:":C$=CHR$(0)
120 FOR I=0 TO 17: GET#1,A$,B$:NEXT
130 CLOSE 1
140 H=ASC(A$+C$)+256*(ASC(B$+C$))
150 IF H=>LL THEN GOTO 170
160 PRINT "SPAZIO SU DISCO INSUFFICIENTE!":END
170 REM SUCCESSIVE ISTRUZIONI
```

Filippo Masia
Sassari

Commodore 128

***187 Nuovi tasti funzione -** Tutti sanno che con l'istruzione Basic KEY del C-128 è possibile attribuire particolari significati ai tasti funzione, in modo che premendoli vengano effettuate operazioni diverse da quelle standard. Non tutti sanno però che è possibile attribuire nuovi significati anche ai tasti HELP, RUN, e alla combinazione di tasti SHIFT RUN/STOP.

Le definizioni dei tasti ridefiniti vengono memorizzate nell'area di memoria compresa fra l'indirizzo $1000 e $10FF. Per ridefinire per esempio la combinazione di tasti SHIFT RUN/STOP in modo che esegua solo il comando RUN:, è sufficiente alterare i byte che vanno dall'indirizzo $103F a $1047. Per effettuare questa operazione è necessario abilitare il programma monitor residente del C-128. Con il comando M 103F 1047 vengono visualizzati i consueti comandi dL"* e RUN in

Input/output rivela ogni mese ai suoi lettori routine di programmazione per C-64, C-128 e Amiga. Avete in mente un'idea particolare, una routine, un utile stratagemma di programmazione, avete sviluppato un breve programma o in generale pensate che le Vostre conoscenze possano interessare l'utenza Commodore? Allora scrivete, e inviate gli eventuali listati stampati su carta bianca o salvati su disco a:

COMMODORE GAZETTE
Input/output
Via Monte Napoleone,9
20121 - Milano

formato ASCII. Se ora si cambiano questi byte in modo da ottenere:

```
00 00 00 00 52 55 4E 3A 0D
```

e si esce dal monitor, la combinazione di tasti corrisponde ora al comando RUN:. Ridefinendo altri tasti e salvando su disco l'area di memoria delle definizioni, è sempre possibile riabilitarli tutti caricando un semplice file da disco. Da monitor il comando per salvare quest'area di memoria è:

```
S "TASTI",8,1000,1100
```

mentre da Basic si può usare l'istruzione seguente:

```
BASVE"TASTI",B0,P4096 TO P4351
```

Per ricaricare il file si digiti il comando:

```
BLOAD"TASTI"
```

Fabrizio Lupano
Bergamo

Commodore Amiga

***188 Numerazione delle righe nei testi -** Certe volte sarebbe comodo che un file di testo avesse le righe numerate. Per esempio, quando si compila un programma in C, e il compilatore restituisce degli errori, li evidenzia indicando il numero delle righe del sorgente nelle quali si sono verificati. Ma il file sorgente, come qualunque altro testo digitato tramite un normale editor, è privo di numerazione di riga. Di solito, quindi, si è costretti a scorrere il listato del sorgente su carta cercando con pazienza le righe incriminate. Ma già con programmi di media grandezza questa ricerca si presenta sempre lunga e tediosa. L'ideale sarebbe avere una copia del file sorgente con la numerazione di riga.

Per ottenerlo da un normale file di testo, ci viene incontro il comando SEARCH impartibile da CLI. Supponiamo che il file di testo per il quale si desidera numerare le righe si chiami *testo.c* e che sia memorizzato nella directory selezionata. Se si digita la seguente linea comando:

```
SEARCH > RAM:testoNume.c testo.c SEARCH ""
```

sul RAM disk viene memorizzato il file *testoNume.c*, che contiene le stesse righe di testo del file *testo.c*, ma numerate. Si tratta di un'operazione tanto semplice quanto efficace, che sarà certamente d'aiuto a molti utenti dell'Amiga.

***189 Programmare con più efficienza -** Certe volte si scoprono piccoli artifici che permettono di rendere la vita del programmatore più semplice. Ve ne propongo due che potrebbero farvi risparmiare del tempo.

Programmando in C o in Assembly, capita spesso che sia necessario accedere alla totalità dei file INCLUDE per scovare la definizione di una struttura. Oppure, che si desideri cercare una funzione all'interno di un gruppo di file sorgente per verificare se nel proprio programma la si sta impiegando correttamente. Per svolgere questo genere di ricerche senza perdere troppo tempo, conviene servirsi del comando SEARCH, che molti utenti evitano di usare per la sua inconsueta sintassi. Se per esempio si vuole estendere la ricerca della parola "StringInfo" a tutte le directory dei file INCLUDE presenti sul disco del nostro compilatore, è sufficiente selezionare la directory include, o asm se il compilatore è un assembler, e impartire il seguente comando:

```
RUN SEARCH > RAM:ricerca SEARCH "StringInfo" ALL
```

e proseguire con altri lavori. Quando il processo in background ha terminato, basta semplicemente impartire il comando:

```
TYPE RAM:ricerca
```

per controllare se la ricerca ha dato i frutti sperati.

Il secondo trucchetto per coloro che programmano assiduamente, ma interessante anche per il generico utente, riguarda una particolarità dell'interfaccia linea comando CLI. Quando si digita un comando che impiega un certo tempo per restituire i suoi risultati, non è necessario aspettare che finisca per digitare altri comandi. Si può infatti procedere subito a impartirne altri anche se il prompt non è ancora apparso. Il CLI li eseguirà uno dopo l'altro, e senza bisogno di alcun processo in background (comando RUN) o file batch (comando EXECUTE).

Se per esempio si deve compilare, sottoporre alla fase di link e infine eseguire un programma nel quale si crede non vi siano errori, si possono digitare in rapida successione le tre linee comando che seguono:

```
CC prog
LN prog -lc
prog
```

e occuparsi d'altro. Il CLI eseguirà questi tre comandi in sequenza, e con un po' di fortuna si potrebbe tornare trovando il proprio programma compilato che funziona regolarmente. ■

SCHEDE ACCELERATRICI PER UN AMIGA "TURBO"

Tutti i metodi hardware e software per trasformare l'Amiga in una fuoriserie capace di prestazioni insospettabili: dalle schede ad alta velocità 68020 e 68030 al Processor Accelerator della CMI, dall'uso di un hard disk ai vantaggi della memoria a 32 bit

di Matthew Leeds

Velocità, potenza, accelerazione. L'irrompere dell'adrenalina mentre la velocità arriva a livelli mai raggiunti prima: è una sensazione che abbiamo provato tutti correndo in auto o nelle discese senza freni in bicicletta.

Ma anche chi possiede un computer vorrebbe provare lo stesso brivido. Riuscite a immaginare un foglio elettronico che esegue tutti i calcoli istantaneamente? Un'immagine 3D che viene elaborata in pochi minuti anziché in giorni? Un disegno che può essere caricato nel giro di qualche secondo? E potremmo continuare a lungo: vi sono centinaia di processi che dovrebbero diventare più veloci.

Ma ogni aumento di velocità ha il suo prezzo. Proprio come accade per un'automobile: più veloce volete che vada e più dovete spendere. Fortunatamente i metodi per migliorare le prestazioni del proprio Amiga sono estremamente vari e di conseguenza vengono proposti a prezzi altrettanto diversi.

Le schede acceleratrici

Se il prezzo non è un problema, allora il meglio del meglio in fatto di supervelocità sono le schede 68020/68030 della CSA o della Finally Technologies. Al gradino più basso della scala, in fatto di costi, troviamo invece il Processor Accelerator della CMI, una nuova versione ad alta velocità del 68000. Altre strategie per il miglioramento della velocità includono l'aggiunta di un hard disk, di memoria extra, o l'uso di un programma cache per la gestione del disco, che grazie alla creazione di appositi buffer accelera lo scambio dei dati. Esaminiamo uno per uno tutti questi sistemi.

L'Amiga è dotato di un microprocessore Motorola 68000 che funziona a 7,16 MHz. Il 68000 legge e memorizza i dati in gruppi da 16 bit, e il bus dati dell'Amiga è stato progettato tenendo conto questa caratteristica. La RAM di sistema, i chip ROM, i chip grafici, i chip di input/output... l'intero sistema, insomma, deve funzionare nell'ambito di queste specifiche: una frequenza di clock di 7,16 MHz e un bus dati a 16 bit.

I microprocessori Motorola 68020 e 68030 fanno invece uso di un bus dati a 32 bit. Elaborare dati organizzati in gruppi da 32 bit è più rapido e inoltre questi microprocessori possono funzionare a una frequenza maggiore dei consueti 7,16 MHz. L'unico problema è che non si può togliere la CPU del proprio Amiga e sostituirla con un'altra come se si cambiasse un pezzo del motore.

A parte il fatto che i pin del 68020 e del 68030 sono diversi da quelli del 68000, vi sono anche altri problemi. Infatti, una CPU a 32 bit ha bisogno del supporto di alcune componenti elettroniche aggiuntive per operare nell'ambito di un sistema a 16 bit: in pratica, per poter cambiare la CPU si deve acquistare un'intera scheda.

La sola aggiunta di una CPU a 32 bit non ha comunque l'effetto di aumentare la velocità di esecuzione delle applicazioni. Infatti, benché la maggior parte delle schede CPU a 32 bit per l'Amiga funzionino a una frequenza di clock di 14 MHz o anche di più, devono comunque rallentare a 7,16 per accedere al bus dell'Amiga. I costruttori delle schede hanno adottato quindi diverse strategie per evitare di dover accedere al bus dell'Amiga, quando è possibile.

La prima, e la più ovvia, è fornire alla nuova CPU una memoria a 32 bit. A questa memoria si può accedere in lettura e scrit-

tura utilizzando gruppi di 32 bit e funziona alla stessa frequenza di clock della nuova CPU. L'aggiunta di memoria può essere effettuata sulla stessa scheda che contiene la nuova CPU, oppure può essere ottenuta con un collegamento a un'opportuna scheda di memoria. I chip di memoria, ovviamente, devono essere all'altezza della nuova CPU, e questo richiede l'uso di chip RAM più costosi di quelli che in genere sono presenti all'interno dell'Amiga.

Una seconda tecnica è spostare tutte le routine del ROM Kernel nella memoria a 32 bit. Dal momento che quasi tutte le applicazioni si servono di queste routine, il loro spostamento nella memoria a 32 bit permette alla nuova CPU di funzionare a piena velocità. Vediamo come si può mettere in pratica quest'idea.

1) Usando una MMU (Memory Management Unit, ovvero "unità di gestione della memoria"), un coprocessore può effettuare lo spostamento richiesto e quindi ricostruire la mappa degli indirizzi di chiamata delle routine perché puntino alle nuove locazioni. La MMU protegge la memoria a 32 bit da eventuali tentativi di sovrascrittura da parte delle applicazioni. A questo scopo, tranne che con il 68030, viene usato spesso il coprocessore Motorola 68851 (il 68030, infatti, ha già una MMU incorporata).

2) Si può creare una ROM a 32 bit personalizzata che contenga le routine del ROM Kernel e costruire la mappa degli indirizzi direttamente per questa ROM. Il vantaggio di tale procedura è che non si spreca preziosa RAM, lo svantaggio è che una ROM personalizzata a 32 bit è piuttosto costosa da costruire.

Aggiungere un 68020 o un 68030 senza aggiungere parallelamente anche una memoria a 32 bit non vale assolutamente la spesa. Il miglioramento nella velocità è soltanto del 30 per cento, e per arrivare a un risultato di questo livello ci sono altre tecniche decisamente meno costose. D'altra parte una scheda 68020/68030 presenta un altro vantaggio: offre il supporto per un coprocessore matematico come il 68881 o il 68882, ovvero coprocessori che effettuano i calcoli in virgola mobile via hardware anziché via software. Ma è un vantaggio relativo, dal momento che, come si vedrà in seguito, ci sono altri modi per offrire supporto a un coprocessore matematico. Quindi, un miglioramento nella velocità veramente significativo si ottiene soltanto quando vengono utilizzati una CPU 68020/30, una memoria a 32 bit e un coprocessore che esegue i calcoli in virgola mobile via hardware.

Il problema con un coprocessore matematico in virgola mobile è che il software deve tener conto della sua presenza. Se un'applicazione non prevede di dover sfruttare un 68881, averne uno a disposizione nel sistema non serve assolutamente a nulla. Ora come ora, soltanti pochi programmi di ray tracing (simulazione d'ambiente) sfruttano le capacità di calcolo del coprocessore matematico.

È difficile dare un giudizio meditato sull'utilità e l'efficienza di una scheda che comprende 68020 e 68881, quando ci sono così pochi programmi che ne fanno un uso esplicito. Esistono alcuni benchmark (programmi di valutazione) scritti appositamente per stimare l'efficienza di questa combinazione, ma il test decisivo è sempre l'impiego delle normali applicazioni. Con il programma *Sculpt-3D*, il rendering di un'immagine di prova eseguito con e senza la coppia 68020 e 68881 (con e senza la memoria RAM a 32 bit) ha dato i seguenti risultati:

- A2000 di serie: 1,00
- 020/881 senza RAM: 1,25
- 020/881 con RAM: 4,50

Ai risultati ottenuti con un Amiga 2000 di serie è stato assegnato un valore unitario, e in proporzione sono stati calcolati i valori da assegnare alla scheda coprocessore. Altre prove hanno messo in luce miglioramenti nella velocità di oltre dieci volte con applicazioni generiche, e di oltre cento volte con programmi di

prova destinati in modo specifico a valutare la capacità di calcolo in virgola mobile.

Le principali aziende che propongono questo tipo di prodotti sono tre: la Computer Systems Associates, la Finally Technologies e la Commodore. La CSA e la Finally offrono entrambe schede dotate di 68030 o di 68020, in combinazione con un coprocessore matematico (68881 o 68882). Mettono inoltre a disposizione questi sistemi con diverse frequenze di clock, fino a un massimo di 25 MHz, e diversi tipi di memoria a 32 bit. La scheda Commodore non è ancora disponibile per il pubblico, ma quando verrà messa in commercio conterrà anche una versione del sistema operativo Unix. In essa sono previsti da 2 a 4 MB di memoria a 32 bit e una MMU.

Un 68000 più veloce

Se il costo della coppia 68020/68881 più la memoria a 32 bit è troppo alto per le vostre finanze, forse potreste prendere in considerazione il Processor Accelerator della Creative Microsystems: il PA consiste in un 68000 a 14,32 MHz montato su una piccola scheda che si installa nello stesso zoccolo utilizzato dalla CPU dell'Amiga. La scheda comprende anche l'alloggiamento per un 68881 (vedere foto nella pagina precedente). Il PA non offre la stessa stupefacente accelerazione delle combinazioni descritte prima, ma è più o meno all'altezza della coppia 68020/68881 senza la memoria a 32 bit, e il suo prezzo è decisamente inferiore. Dal momento che un 68000 non può accedere direttamente al coprocessore matematico, l'accesso al 68881 viene ottenuto come se quest'ultimo fosse un dispositivo di I/O, ricorrendo alle speciali librerie matematiche FFP messe a disposizione dalla nuova versione 1.3 dell'AmigaDOS.

Facendo alcune prove, abbiamo scoperto che la frequenza di clock del 68881 ha scarso effetto sui tempi di esecuzione dei programmi: frequenze di 12, 16 e 20 MHz hanno messo in luce aumenti di velocità trascurabili (gli esperimenti sono stati effettuati sempre con un programma di benchmark). Il motivo è che la maggior parte del lavoro, quando si fa uso di un 68881, consiste nell'impostarlo per eseguire le operazioni richieste, e purtroppo queste operazioni preliminari vengono eseguite alla velocità della CPU. Le funzioni aritmetiche in precisione singola sono addirittura più veloci quando vengono elaborate via software a 14 MHz!

Il complesso delle operazioni necessarie per impostare il 68881 in modo che possa elaborare via hardware funzioni in precisione singola, richiede un tempo decisamente superiore a quello che poi si risparmia nell'esecuzione dei calcoli. Di conseguenza, la velocità di calcolo via software riesce ancora a essere competitiva. Si noti comunque che il programma benchmark si limita a confrontare la velocità d'esecuzione media delle quattro operazioni aritmetiche fondamentali.

La situazione si capovolge quando l'accuratezza di calcolo arriva alla doppia precisione: le quattro operazioni fondamentali sono senza eccezioni più veloci quando vengono eseguite direttamente via hardware (il tempo d'attesa diminuisce di circa il 20 per cento a parità di CPU). E quando si arriva alle funzioni trascendenti in doppia precisione, la diminuzione del tempo d'attesa può superare il 90 per cento (vedere la tavola a fondo pagina).

Attualmente non c'è nessuna applicazione scritta specificamente per sfruttare fino in fondo un 68881 impostato come un dispositivo di I/O. Si è molto parlato di vantaggi e svantaggi dei programmi che utilizzano le librerie matematiche FFP dell'AmigaDOS V1.3, ma il problema, fondamentalmente, è quello dell'uovo e della gallina: i programmatori sono riluttanti a fare un lavoro supplementare per creare applicazioni che possano sfruttare un eventuale 68881 configurato come un dispositivo di I/O finché non aumenta la quantità di dispositivi di questo tipo installati e venduti, e da parte loro gli utenti non se la sentono di affrontare la spesa finché non ne vedono chiaramente i vantaggi.

L'orientamento della Creative Microsystems è quello di offrire non solo un 68881 configurato come un dispositivo di I/O, ma anche un 68000 ad alta velocità: probabilmente questo incoraggerà molti utenti all'acquisto. Dal momento che il 68881 è opzionale sul Processor Accelerator e quindi può essere aggiunto in qualunque momento, a discrezione dell'utente (e tenendo anche conto che il PA offre un sensibile miglioramento di velocità per molte applicazioni senza che sia necessario aggiungere il coprocessore matematico), è probabile che in futuro vedremo aumentare sempre più il numero degli Amiga predisposti per il 68881 e di conseguenza i programmatori potrebbero decidersi a inserire nel loro software la necessaria compatibilità.

Gli hard disk

Aggiungere un hard disk al proprio Amiga non accelera necessariamente il rendering nei

Benchmark per le funzioni aritmetiche

Precisione	Software (senza 881)		Hardware (con 881)	
	7 MHz	14MHz	7 MHz	14 MHz
Doppia	94,78 sec	78,04 sec	68,10 sec	60,42 sec
Singola	61,08 sec	49,62 sec	59,34 sec	52,96 sec

programmi 3D, ma fornisce tutto lo spazio che serve per immagazzinare i risultati ottenuti. Inoltre diminuisce i tempi necessari per caricare le applicazioni, permette di gettare via quell'ingombrante scatola piena di dischi che occupa la scrivania e rende più rapida la correzione dei testi da parte di un word processor che possieda uno spelling checker.

Per rendersi conto di quanto un hard disk può migliorare le prestazioni dell'Amiga basta prendere in considerazione i dati che si ottengono con *DiskPerformance*, un programma di benchmark per l'Amiga progettato per valutare l'efficienza dell'I/O su disco. Nella tavola presentata in questa pagina sono elencati i risultati ottenuti in due successive prove: la prima con un normale disco appena formattato e la seconda con un controller HardCard della Great Valley Products collegato a un hard disk Quantum 40S da 40 MB. L'hard disk utilizzava il nuovo Fast File System.

Come si può vedere, con l'hard disk c'è una notevole differenza di velocità, ma un arido elenco di numeri rende piuttosto difficile afferrare con precisione l'ordine di grandezza di questa differenza. Ci aiuteremo con un esempio. Questo articolo ha richiesto meno di due secondi per essere trasferito su hard disk: praticamente il tempo necessario per premere il tasto e rilasciarlo. Aprire una nuova finestra CLI o Shell richiede circa un secondo dal momento in cui si preme la combinazione di tasti (hotkey) impostata per questo scopo. In pratica non ci sono più tempi morti quando il filing system dell'Amiga cerca e carica i file da disco.

Risultati ottenuti con il disk drive

Creazione:	creati 0 file/sec
Cancellazione:	cancellati 1 file/sec
Lettura directory:	38 voci al secondo
Ricerca e lettura:	17 ricerche/letture al secondo

Velocità di lettura/scrittura

Buffer	*Velocità di lettura*	*Velocità di scrittura*
512 byte	11702 byte/sec	4993 byte/sec
4096 byte	12365 byte/sec	5100 byte/sec
8192 byte	12423 byte/sec	5130 byte/sec
32768 byte	12483 byte/sec	5150 byte/sec

Risultati ottenuti con l'hard disk

Creazione:	creati 14 file/sec
Cancellazione:	cancellati 37 file/sec
Lettura directory:	92 voci al secondo
Ricerca e lettura:	106 ricerche/letture al secondo

Velocità di lettura/scrittura

Buffer	*Velocità di lettura*	*Velocità di scrittura*
512 byte	59578 byte/sec	28493 byte/sec
4096 byte	201649 byte/sec	124830 byte/sec
8192 byte	262144 byte/sec	137970 byte/sec
32768 byte	327680 byte/sec	174762 byte/sec

Il floppy accelerator

Se avete aggiunto ulteriore memoria RAM al vostro sistema (ma anche se non l'avete fatto) può essere interessante avere qualche notizia sul *FACC* della ASDG. Il *FACC* (floppy accelerator) è un programma cache "intelligente" che migliora l'accesso ai disk drive. Mantiene in memoria gli ultimi file letti dal nostro disk drive, combinando le caratteristiche migliori di un RAM disk con quelle del comando ADDBUFFERS e con un'eccellente interfaccia utente.

Tutto questo significa che, se si ha memoria extra a disposizione, si può usare il *FACC* come una specie di RAM disk intelligente, che mantiene il "ricordo" dei file che sono stati richiamati più di recente e di quelli che si usano più spesso, e li mantiene in un buffer (cache) allocato dinamicamente in RAM. Cerchiamo di spiegarlo meglio con un esempio.

Quando si ricorre al CLI per chiamare una directory dal disco, si utilizza il comando DIR dfX: dove X rappresenta il numero del disk drive da cui si richiama la directory. L'AmigaDOS deve recuperare il comando DIR dalla subdirectory C del disco sistema (in genere si tratta del disco del *Workbench* con cui è stato effettuato il boot), e quindi legge e visualizza la directory desiderata. Se si desidera rivedere una seconda volta la stessa directory, l'intero processo dev'essere ripetuto, cioè si deve attendere che venga localizzata la subdirectory C, che venga letto il comando DIR, che la directory venga letta, ordinata e infine visualizzata: una notevole perdita di tempo.

Ricorrendo invece al *FACC*, il comando DIR e la directory vengono immagazzinati in RAM in appositi buffer, e la seconda volta che se ne richiede la visualizzazione, la directory appare non appena si finisce di digitare il comando. Ovviamente il *FACC* riaccede al disco qualora sia stato compiuto un accesso in scrittura che ha modificato l'attuale directory.

È soltanto un esempio banale, e se si limitasse a quest'operazione, il *FACC* non sarebbe altro che una

copia carbone del comando ADD-BUFFERS. Ma il *FACC* possiede altre caratteristiche: esegue l'allocazione del buffer nella fast RAM (ovvero l'ulteriore memoria aggiunta ai primi 512K, cioè alla chip RAM) e non nella chip RAM, inoltre permette di variare le dimensioni del buffer in ogni momento e anche di eliminarlo del tutto, e tiene conto di tutte le volte che presta la sua assistenza e di quanta memoria sta impiegando in ogni momento.

L'ultima versione disponibile, *FACC II*, ha anche altre caratteristiche: riesce a distinguere fra dati e informazioni nelle directory, e dà la precedenza alle informazioni. Ciò significa che *FACC II* individua i file più rapidamente. Possiede inoltre particolari capacità di gestione della chip RAM. Se si sta usando *FACC II* e questo impiega un buffer da 128K, e si tenta di far partire un'applicazione che richiede 100K di memoria, ma in quel particolare momento nel sistema ci sono soltanto 50K liberi, *FACC II* cede una parte della memoria occupata dal suo buffer perché l'applicazione possa essere caricata senza problemi. E tutto senza l'intervento dell'utente.

Ci vorrebbe ancora molto spazio per elencare tutte le altre notevoli caratteristiche di *FACC II*; ci limitiamo a sottolineare che la ASDG ha pubblicato questo programma senza protezioni contro la copiatura e che segue una politica molto generosa per quanto riguarda i miglioramenti che vengono via via apportati al programma. Ogni nuova versione viene infatti fornita gratuitamente a tutti gli acquirenti della versione originale. Se non si possiede un hard disk, non si può assolutamente rinunciare anche al *FACC*: chi farà la prova, si renderà conto che la differenza è stupefacente.

Il Fast File System

Infine, per chi ha uno spirito avventuroso e la mentalità dell'hacker, c'è un altro modo per ottenere la sospirata accelerazione: si può usare il Fast File System non solo con l'hard disk, ma anche con i disk drive. Benché la Commodore non fornisca ancora ufficialmente un supporto per l'uso del FFS con disk drive (sarà disponibile con la versione 1.4 dell'AmigaDOS), ci si può provare lo stesso.

Il problema principale è che l'AmigaDOS non può rilevare la sostituzione del disco, perché si aspetta che il FFS venga utilizzato soltanto su supporti magnetici non estraibili, come appunto l'hard disk. Di conseguenza può succedere che il FFS rovini un disco scrivendovi dati destinati al disco precedente, senza accorgersi che il contenuto del disk drive nel frattempo è cambiato.

La versione 1.2 dell'AmigaDOS ha visto l'introduzione del comando DISKCHANGE, che era destinato principalmente agli utenti che utilizzavano disk drive da 5,25". Ma il comando può benissimo essere utilizzato per informare il Fast File System che il disco presente nel disk drive è cambiato. Ogni volta che si cambia il disco si deve quindi impartire il comando DISKCHANGE: basta dimenticarsene anche una volta sola e si può danneggiare il disco. La sintassi del comando DISKCHANGE è:

```
DISKCHANGE dfX:
```

dove X rappresenta come al solito il numero del disk drive nel quale è stato sostituito il disco.

Grazie al Fast File System è possibile immagazzinare sullo stesso disco una quantità di dati superiore del 4,9 per cento rispetto a quella consueta. Si riesce inoltre a richiamare da disco le directory più rapidamente. Vengono un po' accelerate anche le operazioni di lettura e scrittura (non molto, a causa delle limitazioni dell'hardware). Una differenza veramente notevole, con questo sistema, viene messa in luce quando si cerca di accedere a due o più disk drive diversi contemporaneamente. Se per esempio si stanno salvando alcuni dati su disco in df0:, e si vuole simultaneamente copiare qualcosa dal RAM disk a df1:, non ci sarà praticamente nessun rallentamento nelle due operazioni rispetto al loro ritmo consueto. Esattamente il contrario di quanto sarebbe avvenuto con il vecchio filing system. Si noti invece che realizzando i due accessi sullo stesso disco si ottiene ancora la consueta immediata diminuzione delle prestazioni.

Le operazioni che si devono svolgere in pratica per impiegare questo "trucco" tecnico, sono le seguenti: per prima cosa si deve trasferire il file "FastFileSystem" nella directory L: di sistema, poi si devono aggiungere le seguenti definizioni nella propria MountList di sistema (si trova nella directory DEVS:).

```
/* Definizioni per configurare i normali
df0: e df1: come ff0: e ff1:, ovvero per
trasformarli in dispositivi logici in
grado di avvalersi del Fast File System.
*/

ff0: Device = trackdisk.device
     FileSystem = l:FastFileSystem
     GlobVec = -1
     Mask = 0x7FFF
     Unit = 0
     Flags = 1
     Surfaces = 2
     Mount = 1
     BlocksPerTrack = 11
     Reserved = 2
     PreAlloc = 11
     Interleave = 0
     LowCyl = 0 ; HighCyl = 79
     Buffers = 5
     BufMemType = 3
     DosType = 0x444F5301
#
ff1: Device = trackdisk.device
     FileSystem = l:FastFileSystem
     GlobVec = -1
     Mask = 0x7FFF
     Unit = 1
     Flags = 1
     Surfaces = 2
     Mount = 1
     BlocksPerTrack = 11
     Reserved = 2
     PreAlloc = 11
     Interleave = 0
     LowCyl = 0 ; HighCyl = 79
     Buffers = 5
     BufMemType = 3
#
```

Ora si possono impartire da CLI i seguenti comandi:

```
MOUNT ff0:
MOUNT ff1:
```

Et voila! Adesso ci sono due disk drive FFS nel sistema. Naturalmente, si devono ancora formattare i dischi in modo FFS. Per eseguire questa operazione, s'inserisce un disco vuoto nel disk drive ff1: e si digita:

```
SYS:SYSTEM/FORMAT DRIVE FF1: NAME
''FFS Floppy!'' NOICONS FFS
```

Una volta completata la formattazione del disco si può finalmente crearvi una directory e copiarvi i file. D'ora in poi ci si deve riferire al disk drive come ff1:, non df1:. Per copiare qualcosa su un disco FFS, per esempio utilizzando il proprio disk drive ff1: (il disk drive esterno per i possessori dell'Amiga 500 e 1000), si deve digitare:

```
COPY MyFile TO ff1:
```

È sempre possibile utilizzare i disk drive anche con i dischi vecchio stile. Basta eseguire l'accesso servendosi delle vecchie denominazioni df0: e df1:.

Quando un disco si trova nel disk drive ff1:, il comando INFO avrà come risultato l'affermazione che il disco nell'unità df1: "non è un disco DOS", cosa verissima, dal momento che è un disco FFS. Nella riga di testo per il disk drive ff1: verrà correttamente mostrato il nome del vostro disco. Se si tenta di accedere a un disco FFS utilizzando il nome dfX:, si ottiene ancora il requester "Not a DOS disk". Non si deve fare altro che cancellare la scritta e ritentare con ffX:, e lo stesso vale se si tenta di accedere a un disco non-FFS tramite il dispositivo logico dfX:. Nota bene: il solo modo per sbarazzarsi dei due nuovi disk drive FFS è di impartire il reset alla macchina o di spegnerla e riaccenderla.

Il Fast File System *non dà nessun avvertimento* se si estrae troppo presto il disco dal disk drive (per intenderci, quando il led è ancora acceso). Nelle operazioni di scrittura, per di più, il FFS ha la tendenza a far spegnere il led del disk drive per pochi secondi, e a farlo poi riaccendere per circa un secondo. Il FFS, infatti, ripone le informazioni nei suoi buffer e se per un po' di tempo non c'è attività, trasferisce i buffer su disco. Il FFS, inoltre, in seguito al comando INFO restituisce informazioni riguardo allo stato di protezione del disco non attendibili, che non devono quindi essere prese in considerazione. Lo conferma anche la stessa Commodore, la quale dichiara che il FFS non possiede informazioni sui dischi estraibili, visto che deve lavorare con quelli fissi.

Per concludere

Questo conclude la mia carrellata sull'alta velocità per l'Amiga. Come accade per qualunque altra macchina, ci sono persone che si accontentano di quanto viene prodotto dalla fabbrica e altre che cercano di spremere ogni grammo di prestazione in più che riescono a ottenere... il mio Amiga, per esempio, potete riconoscerlo dalle rigature che gli ho fatto sui lati a forza di aprirlo! ■

Per ulteriori informazioni contattare direttamente:

Computer System Associates, Inc
7564 Trade Street
San Diego, CA 92121, USA

Finally Technologies
25 Van Ness, Suite 550
San Francisco, CA 94102, USA

Creative Microsystems, Inc
10110 SW Numbus Street
Tigard, OR 97223, USA

Great Valley Products
POB 391
Malvern, PA 19355, USA

ASDG
280 River Road, Suite 54A
Piscataway, NY 08854, USA

LE STAMPANTI MANNESMANN MT 87 E MT 88

Funzionalità e buone prestazioni per una famiglia di stampanti professionali. A casa, ma soprattutto in ufficio, velocità e qualità a un rapporto prezzo/prestazioni decisamente interessante

di Luca Giachino

Veloci, di buona qualità, robuste, comode, leggermente rumorose, sono gli aggettivi che meglio descrivono con un colpo di penna le stampanti MT 87 e MT 88 della Mannesmann Tally. Si tratta di due modelli appartenenti alla stessa famiglia, diversi soltanto per la larghezza del foglio di stampa. La MT 87 stampa su carta non più larga di 10", mentre la MT 88 arriva fino a 16". Sono stampanti destinate principalmente all'utenza professionale, ma l'accessibilità del prezzo le rende appetibili anche per coloro che impiegano il proprio computer nel tempo libero. Il modello che abbiamo messo alla prova è la MT 88, ma le valutazioni che esprimeremo valgono anche per la MT 87, dal momento che a parte la larghezza del foglio di stampa sono perfettamente identiche.

La prima impressione che abbiamo avuto è stata di professionalità e compattezza. Per quanto sia più larga della MT 87, anche la MT 88 non può certo essere definita ingombrante (la MT 88 misura 339 x 608 x 185 mm, mentre la MT 87 misura 339 x 454 x 185 mm). Non vi sono elementi sporgenti e il disegno del cabinet è gradevole. Spicca l'ampio vano frontale, al cui interno sono presenti due guide scorrevoli per l'inserimento dei fogli singoli, una soluzione intelligente, perché permette d'inserire i fogli singoli senza doversi sporgere sulla parte superiore della stampante. Inoltre, il vano permette di appoggiare comodamente il foglio singolo senza preoccuparsi di farlo entrare esattamente nel meccanismo di trascinamento a frizione: un opportuno sistema di presa meccanico, azionabile sia da software sia da console operativa, provvede ad agganciare il foglio e a collocarlo automaticamente in posizione di stampa.

Sempre sul lato frontale, sotto il vano d'inserimento dei fogli singoli, è collocato il connettore per le cartucce d'espansione. È infatti possibile dotare la stampante di cartucce addizionali (ne sono disponibili tre) contenenti ulteriori fonti carattere. Sul lato superiore un'ampia finestra semitrasparente di colore bruno protegge i meccanismi di trascinamento e di stampa. La slitta che ne permette l'apertura non sembra una soluzione particolarmente efficiente, dal momento che un semplice incastro avrebbe reso più agevole l'accesso alla testina di stampa e al nastro inchiostrato. Sulla destra risiede la console operativa, una vera e propria interfaccia utente con il firmware di gestione della stampante (la versione che abbiamo avuto in visione ha la serigrafia in lingua italiana, un particolare che abbiamo notato con piacere). Attraverso i comodi tasti e i led l'utente, oltre a svolgere le normali operazioni d'inserimento e avanzamento carta, può accedere a due programmi interni destinati a predisporre la configurazione permanente (switch software) e alla scelta degli stili di stampa. Attraverso alcune procedure semplici da imparare e da ricordare, l'utente dialoga con la stampante per configurare gli switch e scegliere gli stili, senza bisogno di accedere a scomodi microswitch o di agire via software. La possibilità di controllare le funzioni più importanti della stampante anche attraverso la console operativa rende estremamente rapido qualsiasi intervento si renda necessario.

Alla sinistra del rullo di stampa troviamo una levetta per la selezione del tipo di trascinamento (a frizione o a trattore), e una levetta per aprire la frizione in modo da poter intervenire sulla posizione del foglio singolo. L'interrut-

tore d'accensione si trova sul lato destro della stampante, e risulta facilmente accessibile. Sul retro troviamo infine il connettore dell'interfaccia di collegamento con il computer. Il modello che abbiamo provato è dotato di un'interfaccia parallela Centronics, ma è possibile richiedere opzionalmente l'interfaccia seriale RS-232. L'interfaccia parallela permette di collegare la stampante alla maggior parte dei computer. Nella nostra prova il collegamento è stato effettuato con un Amiga.

Le prestazioni

I dati tecnici relativi alle MT sono piuttosto interessanti, e vale la pena di riportarli per esteso. La velocità massima di stampa, quella in modo draft, è di 200 cps (caratteri per secondo), e la stampa è bidirezionale ottimizzata (gli spazi presenti prima di ogni margine nella riga di stampa non producono l'inutile avanzamento della testina). In un ambiente di lavoro dove grandi quantità di dati devono essere trasferite rapidamente su carta, 200 cps costituiscono già una buona velocità.

La stampa è a impatto con matrice di punti, e la testina è a nove aghi. Sebbene la qualità del carattere con un numero maggiore di aghi sia nettamente superiore, la MT 88 ha il pregio di sfruttare al meglio i 9 aghi di cui dispone, con precisione e rapidità. Oltre al modo draft (matrice del carattere di 9 x 9 punti), è disponibile il modo NLQ (matrice del carattere di 18 x 20 punti). In modo NLQ la velocità di stampa si riduce considerevolmente, attestandosi sui 50 cps. In modo draft, il modo operativo che meglio interpreta la filosofia di questa stampante, sono selezionabili tre diverse densità di stampa: Pica (10 cpi, caratteri per pollice), Elite (12 cpi), Condensed (17 cpi). In modo NLQ sono invece disponibili Pica, Elite, e proporzionale. Come si vede, non ci sono molte alternative, ma si tratta di una scelta coerente con la filosofia che caratterizza queste due stampanti della Mannesmann Tally: essenzialità ed efficienza.

Un altro dato di rilievo per quanto riguarda la stampa a caratteri è il numero di caratteri per linea. Ovviamente è un dato che dipende dal modello e dalla densità di stampa selezionata. La MT 87 stampa 80 cpl (caratteri per linea) in Pica, 96 cpl in Elite, 137 cpl in Condensed. La MT 88, invece, stampa 136 cpl in Pica, 163 cpl in Elite, 233 cpl in Condensed.

I set di caratteri disponibili sono il Set Internazionale, il PC IBM US/World e l'Italico Epson, mentre gli standard che la MT è in grado di proporre sono quelli della IBM Graphics Printer, della IBM Proprinter e della Epson FX-80, tre emulazioni che rendono la MT 88 molto versatile. Inoltre la stampante riconosce tutti i codici di stampa più comuni, e garantisce quindi un corretto funzionamento con la maggior parte del software in circolazione.

La densità di punti nella stampa grafica rappresenta sempre un dato da tenere in attenta considerazione, per il potenziale acquirente, specie se dispone di un computer dotato di ottime capacità grafiche come l'Amiga. La MT 88, per quanto sia dotata soltanto di nove aghi di stampa, dal punto di vista grafico offre prestazioni interessanti. La densità orizzontale, espressa in dpi (punti per pollice), è selezionabile fra sei differenti valori: 60, 72, 80, 90, 120, 240. La densità verticale è selezionabile fra 72 e 216 dpi. Vi sono quindi molte combinazioni possibili, anche se sfruttarle appieno è sempre un problema che coinvolge anche il software di cui si dispone.

Per concludere la serie dei dati tecnici, rimane da sottolineare che sono disponibili diversi stili di stampa. In modo draft è possibile abilitare l'espanso, il nero, il neretto, il sottolineato, il corsivo, l'apice e il pedice, mentre in modo NLQ il neretto non è previsto. È infine interessante ricordare che sia la densità sia lo stile di stampa sono facilmente selezionabili attraverso la console operativa in qualunque momen-

to. Se si deve mandare immediatamente in stampa un testo che non contiene codici di controllo (magari con il comando TYPE e la ridirezione dell'output sulla stampante), ma si desidera variare lo stile o la densità di un paragrafo per evidenziarlo, è sufficiente bloccare la stampa al momento opportuno, intervenire tramite la console operativa, e ripartire.

Prova di stampa

Qualita' di stampa: Draft

Tondo
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Espanso
1234567890
abcdefghijklm
ABCDEFGHIJKLM

Compresso
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Nero
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Neretto
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Sottolineato
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Qualita' di stampa: Near Letter Quality

Tondo
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Espanso
1234567890
abcdefghijklm
ABCDEFGHIJKLM

Elite
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Nero
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Neretto
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Sottolineato
1234567890
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Le caratteristiche

La MT 88 dispone del trascinamento a trattore per i moduli continui, e a frizione per i fogli singoli. In entrambi i casi, l'inserimento della carta è sempre molto agevole e preciso, grazie al sistema di aggancio automatico che abbiamo già illustrato. In particolare, troviamo decisamente utile la possibilità d'inserire i fogli singoli dal vano frontale. Anche senza disporre di un caricatore automatico (disponibile opzionalmente per entrambi i modelli), la stampa su una serie di fogli singoli è facilitata dal sistema di aggancio automatico.

Le dimensioni del buffer di stampa, per una stampante, costituiscono sempre un dato di rilievo, soprattutto per i computer che non sono in grado di gestire lo spool (simultaneous peripheral operations on-line) di stampa in background. In questi casi, infatti, se il buffer è di piccole dimensioni il computer è inutilizzabile per tutto il tempo della stampa. La MT 88 dispone di un buffer da 3K, pari al contenuto di circa una pagina. Non è certo un buffer eccezionale, ma nella stampa di fogli singoli risulta in genere sufficiente. Comunque, se si dispone di un Amiga, e si usano i processi in background del CLI (vedere il comando RUN nel *Manuale dell'AmigaDOS*, edito dalla IHT Gruppo Editoriale), oppure se si dispone di software multitasking, il buffer di stampa non ha grande importanza.

Per quanto riguarda gli switch software, dobbiamo rinunciare a una trattazione completa, che

esula dai fini di questo articolo. Il manuale è comunque chiaro e completo, a questo proposito. Ci limitiamo a evidenziare soltanto alcuni aspetti peculiari: per agire sugli switch software, prima di tutto, occorre spegnere la stampante e riaccenderla con una combinazione di tasti premuti. È necessario un modulo continuo già in posizione di stampa, grazie al quale la stampante guida l'utente stampando tutte le informazioni. Il programma di gestione è sintetico ma studiato in modo intelligente: per prima cosa stampa la configurazione di tutti gli switch per una verifica da parte dell'utente, poi passa in rassegna i cinque gruppi di switch disponibili chiedendo ogni volta se deve effettuare qualche cambiamento. In questo modo l'utente può modificare quello che gli interessa senza doverli passare tutti e 34. Peccato però che sul foglio di stampa appaiano soltanto le sigle degli switch, e non la funzione, cosa che costringe l'utente a consultare ogni volta il manuale in dotazione. In questo processo, la stampante continua a dialogare con l'utente attraverso il foglio di stampa, facendolo scorrere avanti e indietro per mostrare quanto ha stampato. Nel corso dell'operazione si può intervenire sul tipo di emulazione, sul set di caratteri, sulla lunghezza del foglio di stampa e sull'interlinea, sulla barratura dello zero, sul salto della perforazione, sul modo di stampa (draft o NLQ), sui parametri impiegati dall'interfaccia e così via.

La configurazione degli switch viene salvata in una EEPROM (Electronically Erasable Programmable Read Only Memory), in modo che, pur non essendoci una batteria tampone, la configurazione impostata non viene persa quando si spegne la stampante. Si ricordi infine che oltre a questa funzione, la console operativa consente anche di alterare la densità e lo stile di stampa senza dover sostituire il foglio e senza dover spegnere la stampante. È

sufficiente mettere la stampante off-line, eventualmente bloccando la stampa in corso, e interagire opportunamente con i tasti e i led della console operativa. Ovviamente, dal momento che con questo secondo programma interno non è richiesto d'inserire un foglio nuovo, la stampante non comunica con l'utente stampando dati: si è adottato invece una sorta di "linguaggio" basato su particolari beep. Anche se può apparire inconsueto, ci è sembrato un sistema abbastanza pratico che riesce a diventare familiare in brevissimo tempo.

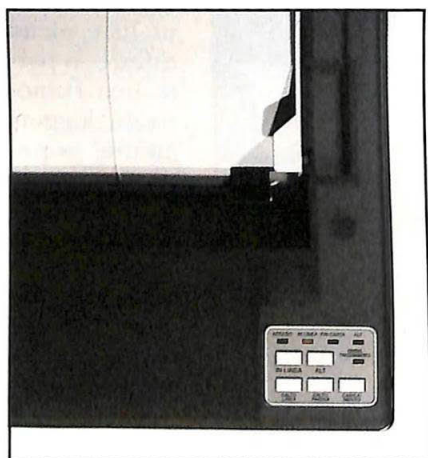
La prova e l'analisi interna

Per effettuare la prova, abbiamo collegato la MT 88 a un Amiga, e ci siamo serviti del driver di stampa Epson. La qualità che abbiamo ottenuto in modo draft e NLQ ci ha favorevolmente impressionato. La leggibilità della stampa è pari se non superiore alla media, la meccanica per l'inserimento dei fogli e il trascinamento si è rivelata robusta e affidabile tanto per il trattore quanto per la frizione. Molte ore di stampa non hanno messo in luce alcun difetto, a parte una rumorosità leggermente superiore alla media.

Gli automatismi di cui dispone e la possibilità di controllare tutte le funzioni attraverso la console operativa, rendono le MT 87 e 88 due stampanti da prendere senz'altro in considerazione per l'ufficio, e da non escludere anche per l'uso non professionale.

Nella stampa grafica la densità è buona, ma ormai questo dato non è più molto significativo per le stampanti professionali, dal momento che per chi intende fare del desktop publishing una stampante laser è l'unica soluzione realmente valida. Quindi, per quanto le MT garantiscano un rendimento accettabile anche nella stampa grafica, inevitabilmente risentono dei limiti intrinseci della stampa a impatto.

Internamente la MT 88 appare

realizzata con cura. È stata dedicata tutta l'attenzione necessaria alla robustezza delle parti meccaniche, aspetto fondamentale per il successo di una stampante nel settore professionale. L'accesso all'interno è agevole, anche se si tratta di un'operazione che per l'utente non dovrebbe certo essere frequente. Notiamo però poca attenzione nei confronti dell'insonorizzazione interna, un dato che comunque ci aspettavamo dato che nella prova la silenziosità si è dimostrata insufficiente. Non sarebbe stato difficile in fase di realizzazione curare maggiormente questo aspetto, che non è certo di secondo piano.

Tiriamo le somme

Le Mannesmann MT 88 e 87 ci sono piaciute. Seguono una precisa filosofia che tende a mettere al primo posto praticità e affidabilità, pur senza lasciarsi andare a soluzioni affrettate, e mantenere al contempo prezzi competitivi. Sono due modelli in grado di soddisfare tanto le esigenze professionali quanto quelle degli appassionati. Non sono stampanti specificamente destinate alla grafica, ma se si desidera soprattutto una buona qualità generale e una soddisfacente stampa dei testi, questi due modelli della Mannesmann Tally meritano senz'altro di essere presi in considerazione. ■

DATI TECNICI

	MT 87	MT 88
Tecnologia di stampa	Matrice di punti a impatto seriale	
Testina di stampa	9 aghi a impatto	
Velocità di stampa	200 cps Draft 50 cps NLQ	
Matrice del carattere (altezza x larghezza)	9 x 9 punti in Draft 18 x 20 punti in NLQ	
Densità di stampa	10/12/17 cpi in Draft 10/12 cpi e proporzionale in NLQ	
Densità grafica	Orizzontale: 60/72/80/90/120/240 dpi Verticale: 72/216 dpi	
Caratteri per linea	80 cpl 96 cpl 137 cpl	136 cpl 163 cpl 233 cpl
Tabulazioni	Verticali e orizzontali	
Larghezza foglio di stampa	Da 4 a 10 pollici	Da 4 a 16 pollici
Lunghezza foglio di stampa	Lunghezza minima 3 pollici (76,2 mm)	
Numero di copie	In media un originale più tre copie (in funzione della grammatura)	
Buffer di stampa	3K	
Trascinamento	Modulo continuo Fogli singoli	
Emulazioni	IBM Graphics Printer IBM Proprinter Epson FX-80	
Interfaccia di serie	Parallela Centronics	
Dimensioni	339 x 454 x 185 mm	339 x 608 x 185 mm
Peso	10,5 kg	15 kg
Livello di rumorosità	Inferiore a 57 decibel	
Prezzo al pubblico	L. 1.092.000 + IVA	L. 1.335.000 + IVA

Optional

Interfaccia RS-232	L. 81.700	
Cartuccia fonti (tre modelli)	L. 93.000	
Caricatore automatico di fogli singoli	L. 367.000	L. 420.000

Importatore

Mannesmann Tally Srl
Via Borsini 6, Corsico (MI)
(tel. 02/4502850)

AMIGA WORKSTATIONS GRAFICHE

Commodore® Computer Center

Pix Computer è un centro specializzato sui sistemi grafici Commodore Amiga. Offerte e preventivi su Workstations grafiche complete chiavi in mano.
Settori CAD 2D/3D - Animazioni 3D/Ray Traling - Digitalizzazioni - Video Broadcast - Desktop Publishing.
INCONTRATEVI CON I NOSTRI ESPERTI NELLA SHOW-ROOM A ROMA, VIA. F. D'OVIDIO 6/C PER INFORMAZIONI, DIMOSTRAZIONI E PREVENTIVI SUI PRODOTTI OPPURE TELEFONATE AI NUMERI: 06/8293507-825731

GENLOCK Mod. PRO-GEN PAL
SEMI PROFESSIONALE
PREZZO LIT. 750.000 IVA inclusa

GENLOCK Mod. MAGNI 4005
PROFESSIONALE BROADCAST
PREZZO LIT. 3.500.000 IVA inclusa

INTERFACCIA GENLOCK Mod. VCG-3

HARDWARE

SISTEMI
AMIGA 500 .... 890.000
AMIGA 500 + MONITOR 1084 .... 1.490.000
AMIGA 1000 SENZA MONITOR 1.250.000
AMIGA 2000 SENZA MONITOR 1.950.000
AMIGA 2000 2 DRIVE 3''1/2 .... 2.190.000

ADD-ONS
ESPANSIONE 512K INTERNA A500 .... TELEFONARE
ESPANSIONE 2MB INTERNA A2000 .... 1.150.000
DISK DRIVE 3''1/2 ESTERNO A500/A1000 .... 250.000
DISK DRIVE 3''1/2 INTERNO A2000 .... 220.000
HARD DISK 20MB EST. A500/A1000 .... 1.190.000
HARD CARD 20MB SCSI A2000 .... 1.190.000
HARD DISK 40MB SCSI PER A2000 .... 1.850.000
HARD CARD 20MB MS/DOS 750.000
HARD DISK 20MB + 1MB RAM A500 .... 1.650.000
SISTEMA A CARTRIDGE DA 12MB REMOVIBILI DELLA KODAK + 5 CARTRIDGE (60MB) 2.450.000
SCHEDA JANUS XT A2000 .... 1.150.000
SCHEDA JANUS AT A2000 .... 1.550.000

SCHEDE ACCELERATRICI
SGS THOMPSON 68000 - THOMPSON 350.000
SGS 68000 + 68881 - THOMPSON 700.000

DIGITALIZZATORI VIDEO
AMIGA - EYE - SCT 130.000
DIGI-VIEW 3.0 PAL - NEWTEK 290.000
GENDER CHANGER PER 500/2000 .... 45.000
DIGI-DROID - NEWTEK 155.000
TELECAMERA B/N 750 LINEE - SECURIT 550.000
STATIVO PROFESSIONALE - TEKNOS 350.000
VIDEON COLORE AUT. NEWTRONIC 390.000
VD - AMIGA COL. TEMPO REALE A500/2000 .... 1.250.000
FRAMER COL. TEMPO REALE - NEWTRONIC 1.250.000
DIGITALIZZATORE AUDIO 130.000
INTERFACCIA MIDI 75.000

GENLOCK
SEMIPROFESSIONALE:
RENDALE 8702 - RENDALE 850.000
BROADCAST:
VCG-3 - M. LAM 1.650.000
RENDALE PROFESSIONAL - RENDALE 1.650.000
MAGNI 4000 - MAGNI 3.900.000

SCANNER
AMISCAN - B.C. 1.590.000

TAVOLETTE GRAFICHE
PENMOUSE KURTA - KURTA 550.000
SERIE IS/ONE 12×12 A4 - KURTA 1.250.000
SERIE IS/ONE 12×17 A3 - KURTA 1.750.000
PENNA A 2 BOTTONI - KURTA 295.000
CURSORE A CROCE - KURTA 295.000
KIT CAVO + SOFTWARE - KURTA 110.000

STAMPANTI A COLORI
P6 PLUS 24 AGHI - NEC 1.850.000
P7 PLUS 24 AGHI - NEC 2.290.000
HQP40 24 AGHI - CITIZEN 1.590.000
4020 INKJET - RANK XEROX 3.500.000
PAINTJET - HEWLETT PAC 3.350.000
HARD COPIER TOYO - TOYO .... TELEFONARE

STAMPANTI LASER
LGQ 3500 - EPSON 3.950.000
NEC LC-890 POSTSCRIPT - NEC 10.500.000
DESKJET (INKJET) - HEWLETT PAC 2.500.000

PLOTTER
DXY-800A - ROLAND 1.600.000
DXY-880A - ROLAND 2.390.000
DXY-885 - ROLAND 3.050.000
DXY-990 - ROLAND 3.990.000
DPX-2000 - ROLAND 11.590.000
DPX-3000 - ROLAND 15.000.000

PALETTE
PALETTE PER AMIGA - POLAROID 3.900.000

SOFTWARE APPLICATIVO

DATABASE
SUPERBASE PERSONAL - CTO 180.000
SUPERBASE PROFESSIONAL - CTO 390.000
ACQUISITION 1.3 - TAURUS 490.000

WORD PROCESSOR
TEXTCRAFT PLUS - CTO 145.000
KINGWORDS - CTO 60.000
PROWRITE - NEW HORIZONS 175.000
EXCELLENCE! - MSS 419.000

DESKTOP PUBLISHING
PAGE SETTER - CTO 210.000
PROFESSIONAL PAGE - GOLD DISK 490.000

SPREADSHEET
LOGISTIX - CTO 120.000
MAXIPLAN 500 - OXXI 220.000
MAXIPLAN PLUS - OXXI 290.000

LINGUAGGI
LATTICE C 4.1 - LATTICE 290.000
PASCAL - METACOMCO 250.000
MACROASSEMBLER - METACOMCO 250.000

C.A.D.
INTROCARD - CTO 94.000
DRAW PLUS - AEGIS 350.000
PROFESSIONAL DRAW - AEGIS 390.000
DYNAMIC CAD - MICROILLUSIONS 690.000
HOME BUILDERS CAD - EASEWARE 350.000
X-CAD DESIGNER - TAURUS 950.000

MODELLATORI 3D
THREE DEMON - MIMETICS 125.000
FORMS IN FLIGHT - MICRO MAGIC 125.000
FORMS IN FLIGHT II - MICRO MAGIC 350.000
VIDEOSCAPE 3D - AEGIS 199.000
MODELER 3D - AEGIS 199.000
SCULPT 3D - BYTE BY BYTE 135.000
SPACE SHUTTLE CLIP ART - BYTE BY BYTE 195.000
TURBO SILVER 3D - IMPULSE 450.000

ANIMAZIONE 2D/3D
ANIMATOR - AEGIS 120.000
PHOTON CELL ANIMATOR - MICROILL 250.000
FANTAVISION - BRODERBUND 95.000
ANIMATE 3D - BYTE BY BYTE 225.000
ANIMATE 3D PROF - BYTE BY BYTE 550.000

GRAFICA PITTORICA
DPAINT - CTO 99.000
PHOTON PAINT - MICROILL 160.000
DIGI-PAINT - NEWTEK 120.000

PRESENTAZIONI GRAFICHE
DVIDEO - CTO 109.000
DELUXE PRODUCTIONS - ELECTR.ARTS 160.000
IMPACT - AEGIS 120.000
THE DIRECTOR - R.A.G. 99.000
LIGHTS, CAMERA, ACTION - AEGIS 115.000
TW SHOW - ZUMA 110.000

VIDEOTITOLATORI
VIDEOTITLER - AEGIS 160.000
TV TEXT - ZUMA 110.000
PROVIDEO PLUS PALL - JDK 419.000
FONTS PRO VIDEO - JDK 65.000
PRO VIDEO FONTS 1 - JDK 290.000
PRO VIDEO FONTS 2 - JDK 290.000
KARA FONTS - KARA 110.000
FANCY 3D FONTS - ACCESS 115.000
VIDEO EFFECT 3D - INNOVISION 250.000
THE CALLIGRAPHER - INTERACTIVE 140.000

UTILITY GRAFICHE
PIXMATE - CTO 90.000
BUTCHER - EAGLE TRHEE 45.000
INTERCHANGE - SINDESIS 110.000
DPRINT - CTO 90.000
DIGI-VIEW - NEWTEK 290.000

MODEM
DIGA! - AEGIS 99.000

MUSICALI
DMCS - CTO 94.000
HOT LICKS - INFINITY 55.000
AUDIO MASTER - AEGIS 125.000
PRO MIDI STUDIO - MIMETICS 250.000
KCS 1.6 - DR.T'S 390.000

VARI
SURGEON - ISM 65.000
FACC II - ASDG 49.000
DOS-2-DOS - CENTRAL COAST 95.000
THE PLANETARIUM - MICROILL 95.000

SOFTWARE INTRATTENIMENTO

PAC-MANIA 25.000
P.B. FOOTBALL 29.000
STARGOOSE 39.000
DALEY THOMPSON'S 25.000
VIRUS 39.000
OUT-RUN 25.000
SORCERY PLUS 25.000
SUB BATTLE SIMULATOR 29.000
ULTIMA IV 49.000
THUNDERCATS 29.000
BOMB JACK 29.000
NEBULUS 25.000
BAD CAT 29.000
ROBBEARY 25.000
PLATOON 39.000
STREET FIGHTER 29.000
WIZARD WARZ 25.000
CYBERNOID 25.000
TRACKER 49.000
IMPOSSIBLE MISSION II 39.000
BETTER DEAD THAN ALIEN 55.000
HYBRIS 45.000
DUNGEON MASTER 55.000
DOUBLE DRAGON 39.000
4×4 OFF ROAD RACE 25.000
ALIEN FIRES 69.000
THE PLANETARIUM 99.000
THE FAERY TALE 89.000
SHADOWGATE 69.000
DEFENDER OF THE CROWN 59.000
HALLEY PROJECT 69.000
DEJA VU 69.000
UNINVITED 69.000
TERROR PODS 49.000
BARBARIAN 55.000
OBLITERATOR 55.000
FLIGHT SIMULATOR 85.000
JET 85.000
SCENERY DISK 7 45.000
SHANGHAI 29.500
BORROWED TIME 65.000
LITTLE COMPUTER PEOPLE 65.000
MINDSHADOW 35.000
TASS TIMES 35.000
GEE BEE AIR RALLY 55.000
ARAZOK'S TOMB 49.000
ARKANOID 75.000

PIX COMPUTER S.R.L.
VIA F. D'OVIDIO, 6C
TEL. 06/8293507-825731
00137 ROMA
COMPUTER & Co.
P. IVA 08309630583

TUTTI I PREZZI SONO IVA INCLUSA COMMODORE È UN MARCHIO REGISTRATO

SALONI ITALIANI: SMAU '88

NOVITÀ COMMODORE? NON CERTO TROPPE

Alla 25ª edizione dello SMAU, la filiale italiana della casa di West Chester non ha presentato reali novità al pubblico, limitandosi a ospitare aziende italiane produttrici o importatrici di software e hardware

di Giovanni Varia

L'ultima edizione del Salone Internazionale per l'Ufficio, lo SMAU '88, si è svolta dal 29 settembre al 3 ottobre alla Fiera di Milano. L'area espositiva è aumentata in maniera rilevante passando dai 94 mila metri quadrati dello scorso anno ai 120 mila di quest'anno e raccogliendo circa 1.660 espositori, oltre 220 in più rispetto all'87.

Il segnale di una certa vitalità è stato confermato anche dai numerosi convegni che hanno fatto da contorno alla manifestazione, da quello organizzato dalla SIP a quello dei dottori commercialisti. Tra i tanti ricordiamo soprattutto quello dell'Assoft, l'associazione per la tutela del software. I relatori hanno affrontato problemi importanti come il fenomeno della duplicazione abusiva, che ha come principale conseguenza la contrazione degli investimenti destinati a ricerca e sviluppo da parte delle software house. Si è parlato anche della netta differenza d'impostazione strategica tra le imprese italiane e quelle europee, e infine della diffusa sfiducia da parte dei potenziali partner stranieri nei confronti delle nostre aziende. Sono intuibili le difficoltà che questo comporta, soprattutto nello scambio di risorse e di know-how. In quest'ottica, è stata sottolineata l'importanza di una legge sul software e di una serie di "alleanze concorrenziali" per affrontare da posizioni vincenti lo scenario del 1992.

La presenza della Commodore

Tradizionalmente, la partecipazione della Commodore Italiana allo SMAU si risolve in una mostra dei nuovi prodotti e nella cessione di quasi tutto il suo spazio espositivo a produttori, importatori o rivenditori specializzati in prodotti software e hardware per le macchine Commodore. E quest'anno le cose non sono andate diversamente, se si eccettua una sola, trascurabile, variazione sul tema: di nuovi prodotti non se ne sono visti.

Il pubblico avrebbe certamente gradito la dimostrazione di qualche novità, e molti tra i partecipanti speravano di avere la possibilità di rivolgere almeno qualche domanda al personale tecnico o di poter avere uno scambio di opinioni sui programmi futuri. Ma alla Commodore hanno ritenuto sufficiente esporre alcune macchine già reperibili nei negozi e affidarne la dimostrazione a rivenditori e produttori esterni.

Sembra che questo vuoto totale sia da attribuire al mancato arrivo di alcuni prodotti che dovevano essere spediti dal quartier generale europeo di Francoforte – non è quindi una responsabilità italiana – ma il dato va comunque rilevato: novità non ne abbiamo viste.

Non possiamo quindi dire di aver finalmente toccato con mano i più interessanti prodotti già presentati ai maggiori saloni statunitensi e già citati dalla stessa *Commodore Gazette* in precedenti articoli, come le schede UNIX e AT, il monitor monocromatico A2024 o le nuove versioni del *Kickstart* e del *Workbench*. L'unica novità di marca Commodore presente al Salone è stata il Videomaster A2995, un sistema semiprofessionale per l'elaborazione video, composto da genlock, digitalizzatore e alloggiamento per tre disk drive A2010. Le caratteristiche tecniche del prodotto sono: ingressi videocomposito e RGB;

uscite videocomposito, RF e RGB più sync; controlli di contrasto, colore, saturazione, dissolvenza e fase sul pannello frontale; genlock escludibile; segnale video in ingresso escludibile; foratura sui colori RGB (chroma-key); commutazione ingressi video 1 o 2; digitalizzatore video a colori con filtratura elettronica dei colori. Il prezzo al pubblico, IVA esclusa, dell'A2995 è di un milione e 200 mila lire.

Il produttore di questo apparecchio, la **TDS**, ha annunciato la futura disponibilità di altri sistemi dello stesso tipo che potrebbero avere come destinazione anche il campo professionale.

Tra le aziende che esponevano programmi d'intrattenimento, sia per chi cerca applicazioni professionali. Va ricordato che la C.T.O. detiene l'esclusiva per il nostro Paese di marchi come Logistix, Portex, Precision, Lifetree, Electronic Arts e Progressive Peripherals & Software. Recentemente è stato siglato un accordo anche con la nota software house francese **Infogrames**. Per gli utenti delle macchine Commodore a otto bit è sicuramente interessante sottolineare che C.T.O. non è più solo sinonimo di software per l'Amiga: nel suo catalogo sono ora disponibili un considerevole numero di titoli per C-64/128, in particolare le ultime novità della Electronic

gestire la contabilità aziendale (conti, sottoconti, clienti e fornitori, IVA, prima nota, libro giornale). Si possono eseguire le stampe dei partitari clienti/fornitori, dei bilanci di verifica e un rendiconto suddiviso in conti patrimoniali e di reddito. Il programma è già predisposto per l'interazione con altre applicazioni riguardanti procedure di magazzino, bolle/fatture e analisi di bilancio. Con *Fatturazione* si viene incontro alle esigenze di bollettazione e fatturazione, offrendo un archivio di clienti e articoli. A richiesta, le stampe possono essere impostate anche su moduli personalizzati. *Gestione negozi* consente la gestione di un esercizio commerciale ed

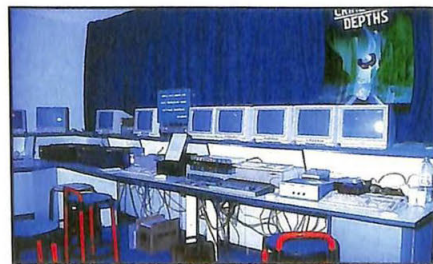
L'esposizione del software importato e tradotto in italiano dalla C.T.O. di Bologna

allo stand Commodore, abbiamo selezionato quelle la cui presenza è apparsa più significativa. Riportiamo qui di seguito una sintetica descrizione delle loro attività, inquadrandole nell'ambito generale del mercato.

Produttori e importatori

La **C.T.O.** ha presentato un'ampia rassegna di programmi originali dedicati all'Amiga e tradotti in italiano. Ormai il catalogo della C.T.O. è piuttosto ampio e può costituire motivo d'interesse sia per chi è alla ricerca di giochi o

Arts e della Infogrames.

La **Atema**, software house specializzata nella realizzazione di software personalizzato per l'Amiga e per sistemi MS-DOS, era presente con sei programmi. *Play Tex*, un pacchetto destinato ai disegnatori tecnici che permette di realizzare tramature di tessuti sullo schermo, elaborandone le componenti caratteristiche. In particolare, c'è la possibilità di evidenziare gli effetti della variazione di ogni singolo componente di un tessuto, e per tale ragione viene consigliato come strumento didattico da utilizzare negli istituti tecnici. *A-Cont* permette di

è diviso in tre routine modulari: magazzino, prima nota e scadenziario. Il programma *Dieta* si propone di aiutare il medico generico o il dietologo, mettendo a sua disposizione uno strumento per gestire l'archivio pazienti e suggerire diete personalizzate. Un'apposita routine compila i menu settimanali facendo le sue

scelte in un campionario che comprende 250 piatti diversi. È anche possibile compilare statistiche e grafici personalizzati. *Ricette* si propone di ottimizzare le scorte di grandi comunità come mense, ristoranti o alberghi. Il programma è in grado di fornire un elenco dei vari piatti che si possono preparare con le scorte disponibili in dispensa (anche in rapporto alle quantità da servire) e di compilare una lista della spesa.

La **Bucolo** ha fatto alcune dimostrazioni dei suoi pacchetti per l'Amiga. *TeleAmigaLive* è destinato all'uso in ambito televisivo e comprende: *TotoLive*, per la gestione dei dati totocalcio; *ElezLive*, per la gestione dei dati degli scrutini elettorali; *WeatherLive*, per illustrare le condizioni meteorologiche; *ReferLive* per la gestione delle consultazioni referendarie; *GameLive*, per giochi d'intrattenimento televisivo. Un'applicazione del tutto diversa è *Kitchen Toolkit*, che ha lo scopo di fornire un semplice e veloce sistema di progettazione a tutti coloro che operano nel settore delle cucine componibili. Il programma consente di selezionare le varie possibilità di arredamento di una cucina – costruita a partire da una serie di oggetti predefiniti – illustrandole così a un eventuale cliente. È anche possibile modificare e/o aggiungere i componenti e ottenere output su carta del progetto finale sia in bianco e nero che a colori. Sulla scia del programma precedente è stato realizzato *ArredalToolkit*, che consente di arredare su schermo ambienti come bagni, salotti e stanze da letto. In collaborazione con la **Interwideo** di Bologna, la Bucolo realizza anche sistemi informativi in grado d'interagire con il pubblico attraverso sensori a pressione destinati a servizi molto vari. La realizzazione di questi sistemi si basa sul pacchetto software *Windomat* che permette di visualizzare su un monitor le informazioni richieste dall'utente tramite un'interfaccia dotata di appositi tasti per la selezione. La Bucolo produce anche un programma di gestione rappresentanti, *AmigaAgent*, e distribuisce infine il software per MS-DOS della **IHS Italia** (*Biblion, Tapeteca, Infosan* e *Paidion*).

La **Hardital** ha presentato in anteprima allo SMAU Zorro Big Blue. Si tratta di un box d'espansione per Amiga 500 e 1000 che consente di utilizzare anche con questi due modelli le schede di espansione realizzate per il 2000. L'organizzazione del box è modulare e comprende tre slot Amiga a 100 pin Zorro 2, tre slot IBM XT compatibili, tre slot IBM AT compatibili, uno slot a 86 pin CPU, un alimentatore da 40 watt, due alloggiamenti per disk drive da 3,5", un alloggiamento per disk drive da 5,25", un alloggiamento per hard disk da 3,5", un alloggiamento per digitalizzatore stereo e interfaccia MIDI.

La **Logitek**, azienda specializzata in videografica e desktop publishing, ha presentato alcune novità di un certo interesse. Un'espansione di memoria interna da 1,8 MB per A500 e A1000 (L. 1.200.000 + IVA); uno scanner manuale per A500 e A2000, che legge con una definizione di 200 dpi 16 tonalità di grigio (L. 700.000 + IVA); una tavoletta grafica Kurta da 8,5" x 11" (L. 1.116.000 + IVA); un'interfaccia per la gestione di videodischi (L. 980.000 + IVA). Vanno anche ricordati Frame Buffer, un digitalizzatore a colori (L. 1.290.000 + IVA), e Splitter, da utilizzare eventualmente in coppia con il Frame Buffer, che serve per migliorare la qualità delle digitalizzazioni e trasforma il segnale da videocomposito a RGB. Infine, per chi fosse interessato, la Logitek è anche un'azienda di servizi (digitalizzazione d'immagini, titolazione di videocassette, trasferimento d'immagini su diapositiva, formazione professionale...).

La **Gread Elettronica** ha realizzato un'apparecchiatura elettronica destinata esclusivamente alle ricevitorie, ideata e costruita per la stampa di schedine Totocalcio, Enalotto e Totip. Il prodotto si chiama E64/13, ed è costituito da un C-64 con disk drive e da una stampante Epson 240 racchiusi in due cabinet dedicati.

La **Programma Idea** ha presentato al pubblico il programma *Amiga Professional*, progettato per permettere l'elaborazione di sistemi Totocalcio, Enalotto e Totip. Si tratta di un prodotto modulare e aperto a eventuali aggiornamenti, che si rivolge sia a privati che vogliono elaborare da soli un proprio sistema di gioco, sia alle ricevitorie che possono così offrire al pubblico un servizio completo. Le caratteristiche principali di questo pacchetto sono: gestione fino a 15 pronostici, inserimento della schedina setti-

manale, colonne condizionate And e Or con consecutività, interruzioni, elaborazione statistica dei dati relativi al sistema con conseguente controllo delle colonne elaborate, accorpamento ottimizzato, riduzione a n−1 del sistema integrale condizionato, stampa di schedine e tabulati.

La **Ideamatix** opera nel campo dei video utilizzando computer Amiga collegati con svariate periferiche. Propone prodotti come espansioni di memoria, schede di rendering, genlock, digitalizzatori, interfacce MIDI, espansori Thru/Out, commutatori Multi In. Nel campo dei servizi si occupa di postproduzione video, realizzazione di logo e installazione di workstation grafiche.

Il software MS-DOS

Per quel che riguarda gli IBM compatibili prodotti dalla Commodore, era presente un discreto numero di produttori e abbiamo assistito a diverse dimostrazioni dei più recenti pacchetti software. Ricordiamo per esempio la **Totobit Informatica** che proponeva *Videobit*, programma per la gestione delle videoteche; la **C.M.B.** presente con *Ore assenza*, programma di gestione e controllo delle ore di lavoro, *Pay*, per l'amministrazione del personale, *Rimborso spese*, per gestione e controllo della nota spese, e *Selemaster*, dedicato alla selezione dei candidati. La **Cared Informatica** era presente con *Spcapc*, package dedicato alla gestione aziendale nel suo complesso, dalla contabilità generale alla gestione magazzino; la **EDS**, infine, presentava *EDS Condomini*, programma destinato agli amministratori di stabili.

Al prossimo anno

Lo SMAU è una manifestazione i cui spazi espositivi sono in continua crescita: rispetto all'ultima edizione l'incremento è stato del 30%. Quello che sorprende è che l'aumento della superficie espositiva non corrisponda a un parallelo incremento dei motivi d'attrazione e a un crescente impegno dei singoli espositori. Sarebbe piaciuto a molti assistere a dimostrazioni dei nuovi prodotti Commodore affidate a tecnici esperti. Si è invece preferita la soluzione più semplice: dimostrazioni a opera di produttori esterni, che in qualche caso possono essere suggestive ed esaurienti, ma che spesso lasciano gli spettatori con troppe domande senza risposta. Chissà, forse il prossimo anno... ■

Per ulteriori informazioni contattare direttamente:

Assoft
Via Cellini, 1
20100 Milano
(tel. 02/55183116)

Atema
Via B. Marcello, 1b
50144 Firenze
(tel. 055/352661)

Bucolo
Via Sottomonte, 5
95030 Pedara
(tel. 095/915265)

Cared Informatica
Via Visconti Venosta, 2
20122 Milano
(tel. 02/5456845)

C.M.B.
Via G. Sismondi, 44
20133 Milano
(tel. 02/714678)

Commodore Italiana
Via F.lli Gracchi, 48
20092 Cinisello Balsamo
(tel. 02/618321)

C.T.O.
Via Piemonte, 7/F
40069 Zola Predosa
(tel. 051/753133)

E.D.S.
P.za S. Agostino, 5
20100 Milano
(tel. 02/8375905)

Gread Elettronica
Rovereto
(tel. 0464/443235)

Hardital
Via Tortona, 12
20100 Milano
(tel. 02/8376887)

Ideamatix
Via Bettini, 2
20128 Milano
(tel. 02/2552875)

IHS Italia
Via Privitera, 3
95038 S. Maria di Licodia
(tel. 095/629083)

Interwideo
Via Vasco Da Gama, 25
40131 Bologna

Logitek
Via Golgi, 60
20133 Milano
(tel. 02/2666274)

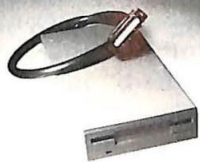
TDS
C.so Domenico Riccardi, 347
80040 Cercola
(tel. 081/7332797)

Totobit Informatica
Via San Pietro, 32
89100 Reggio Calabria
(tel. 0965/594463)

EASY-SOUND
digitalizzatore di suoni e voci per Amiga 500/1000/2000 fino a 22Khz. Completo di microfono e software ricco di controlli, possibili mixaggi, (IFF). **L. 119.000**

MODEM SMARTLINK 1 - 2

V 21 - 22	L. 290.000
V 21 - 22 - 23	L. 319.000
2400	L. 329.000

TELECAMERA B/N ALTA RISOLUZIONE (600 LINEE)
da accoppiare a DIGIVIEW, EASY VIEW, REALTIME, VID VIDEON ecc.
L. 399.000

I NOSTRI DISK DRIVE

DISK DRIVE SLIM, MECCANICA NEC BEIGE

è disponibile:

PER AMIGA 500 3,5 pollici passante **L. 239.000**

PER AMIGA 500 5,25 pollici 40/80 tracce passante **L. 269.000**

PER AMIGA 2000 interno **L. 179.000**

PER COMMODORE 64 OCC118 **L. 239.000**

HARD DISK CARD:
per A2000 in modo MS-DOS (MECCANICA MINISCRIBE, CONTROLLER WESTERNDIGITAL)

20 MB	L. 639.000
33 MB	L. 799.000
40 MB	L. 969.000
20 MB HARD DISK	L. 539.000

SERVIZIO IN FASE DI ATTIVAZIONE:

COMPUTER MAIL (spedizioni computerizzate in tempo reale). Scrivere con espresso per ordini a: ELETTOR - Via Imola, 13 - 20158 Milano - Telefono funzionante da gennaio: 02/3762701. Telefono in funzione (solo alla mattina): 02/3270226

Tutto il materiale è garantito 12 mesi + 7 giorni di prova soddisfatti o rimborsati, ed è in pronta consegna. Noi vendiamo fatti, non parole. I nomi, i marchi e gli stemmi usati in questa pubblicità sono depositati e di proprietà delle menzionate aziende. NEWEL ne è solo il rivenditore, o il distributore, e ringrazia le medesime per l'utilizzo. I prezzi possono variare senza preavviso.

DIGITALIZZATORE DI IMMAGINI IN TEMPO REALE (SEMIPRO)
vari colori, livelli 2-16
L. 179.000
Versione COMMODORE, ATARI ST, AMIGA ecc.

ALCUNE PROPOSTE

KICKSTART 1.3 SU ROM PER AMIGA
(2MB di memoria EPROM)
con possibilità di mantenere il vecchio sistema . . . . **L. 149.000**

PROGRAMMATORE EPROM PER AMIGA
(fino a 512 EPROM) . . . . **L. 239.000**

PROGRAMMATORE EPROM PER CBM 64
(fino a 512 EPROM) . . . . **L. 239.000**

TAVOLETTA GRAFICA . . . . . . **L. 600.000**

GRABBER (tempo reale) . . . . . . **L. 390.000**

INTERA MAINBOARD per trasformare il tuo PC XT in AT (MOTHERBOARD) 6/12 MHz . . . . **L. 568.000**

MOTHERBOARD
XT 4/18/10 MHz . . . . **L. 170.000**

CAVO FLIGHT si può giocare tra due computer (Amiga o PC) a Flight Simulator o ad altri giochi per due . . . . **L. 23.900**

STAMPANTI:

COMMODORE

MPS1200/1250 . . . . **L. 479.000**
MPS 1500 . . . . **L. 549.000**

STAR LC 10 B/N . . . . **L. 510.000**
COLOR . . . . **L. 610.000**

NEC 2200 P6-P7 . . . . **Telefonare**

AMSTRAD . . . . **Tutta la serie**

INOLTRE PER GLI APPASSIONATI POSSIAMO FORNIRE PC XT/AT/386 A PREZZI STRACCIATI (RICHIEDERE IL CATALOGO STAIL ELECTRONIC)

OFFERTE SPECIALI

SERIE DI CAVETTI PER COLLEGAMENTO PC e/o C-64 AL 1084/8833 . . . . **L. 19.000**

INTERFACCIA PARALLELA GRAFICA
PER C-64/128 . . . . **L. 99.000**
SERIALE . . . . **L. 59.000**

PROGRAMMI ORIGINALI PER COMMODORE (THE MANAGER, LOGO, CALC RESULT, GEOS, BASIC) . . . . **L. 39.000**

NASTRI INCHIOSTRO PER 80L . . **L. 9.900**

CARTRIDGE EMULATORE CP/M PER C-64 . . . . **L. 29.000**

INTERFACCIA IEE 488 . . . . **L. 29.000**

FILTRO ANTIDISTURBO PER COMPUTER . . . . **L. 19.000**

REGISTRATORI ORIGINALI PER C-64 . . . . **L. 43.900**

PRONTA CONSEGNA DELLE SEGUENTI MARCHE AI PREZZI MIGLIORI, E CON GARANZIA DELLE CASE FABBRICANTI E INTERNA

AMSTRAD Star NEC Seagate Commodore sinclair ATARI SANYO

NOI TI DIAMO PRODOTTI DI MARCA AL PREZZO DEI PC TAIWANESI

UNA VISITA AL MONDO COMMODORE

Dal 3 al 6 novembre si è tenuta a Filadelfia la mostra "The World of Commodore", organizzata nel Civic Center della città: duemila metri quadrati di spazio espositivo completamente dedicati ai prodotti Commodore

di Pietro Vago e Andrea Bellavita

Il "The World of Commodore" si è tenuto per la prima volta anche negli Stati Uniti. L'edizione canadese che si svolge ogni anno a Toronto è giunta ormai al sesto appuntamento (dall'1 al 4 dicembre) e ha ora un prestigioso gemello a Filadelfia: una grande esposizione dei principali prodotti del mondo Commodore, che per la sua prima edizione ha scelto come sede il maggior centro espositivo della città, il Civic Center dell'Università di Pennsylvania.

Una nota d'ambiente: il salone destinato agli espositori era rivestito da una moquette blu col simbolo della Commodore che sbucava dappertutto. Non mancava anche una suadente voce femminile, che dagli altoparlati scandiva ogni minuto uno slogan: «Only Amiga makes it possible...».

Gli articoli che hanno riscosso il maggior successo di pubblico (a giudicare dalle lunghe attese in coda necessarie per poterli ammirare con i propri occhi) sono stati *Dragon's Lair* della Readysoft, *Movie Setter* della Gold Disk, *GEOS 2.0* della Berkeley Softworks, *Max the Mac Emulator* della Readysoft e *Red Storm Rising* della Microprose. Per quanto riguarda l'hardware, dobbiamo citare subito almeno tre prodotti: Impact A500 hard disk/RAM della Great Valley Products, l'espansione SIN 500-2 per l'A500 della Spirit Technology, e Printer Buffer della Supra Corporation.

Per garantire un quadro quanto più possibile esauriente della mostra, passiamo ora a una panoramica sui vari espositori.

La Commodore

La **Commodore** ha presentato il genlock interno per l'A2000 (destinato al mercato USA), modello A2300. Verrà messo in commercio al prezzo di circa 400 dollari. Sono state inoltre presentate, e a quanto pare sono finalmente disponibili anche per il pubblico, le tanto attese versioni 1.3 di *Kickstart* e *Workbench* nelle seguenti configurazioni: pacchetto software (*Kickstart/Workbench 1.3* e accessori, su disco) a 22 dollari, *Kickstart 1,3* su ROM a 39 dollari. Com'è noto, il *Kickstart 1.3* permette il boot da hard disk, ma i possessori del vecchio modello A2090 non potranno usufruirne senza un'apposita scheda che sarà disponibile entro gennaio. I tecnici della Commodore la definiscono una "slot-waster" in quanto "spreca" uno slot di espansione, quando si sarebbe potuto modificare il vecchio hard disk senza bisogno di nessuna scheda aggiuntiva. Il *Workbench 1.3* contiene, oltre a un nuovo e più efficiente interprete di linea battezzato Shell, alcuni nuovi comandi.

Gli stand e i prodotti

La **Berkeley Softworks** ha presentato la nuova versione di *GEOS*, la 2.0, con la quale è possibile utilizzare il disk drive 1581 da 3,5'' e ottenere più rapidamente le normali funzioni DOS. Si può anche recuperare l'ultimo file cancellato. Nel pacchetto sono inclusi i programmi *geoSpell* (un'applicazione per il controllo ortografico), *Text Grabber* (per prelevare da qualsiasi

word processor testi destinati a *geoWrite*) e infine *geoMerge*. Il pacchetto viene messo in commercio al prezzo di 59,95 dollari. È stato inoltre annunciato un pacchetto comprendente *geoBasic* e *geoProgrammer*, che servirà per scrivere programmi in Basic e compilarli; il Basic sarà però completamente rinnovato soprattutto per ottenere una miglior gestione della grafica. Di prossima uscita è *geoChart*, un semplice programma per la creazione di grafici (istogrammi, grafici a torta e così via).

La **Digitek** ha messo in mostra *Hole in one Miniature Golf* (39,95 dollari), un programma di minigolf su due dischi con cinque percorsi da 18 buche. Gli scenari sono numerosi e la grafica è di buon livello; tra le opzioni disponibili c'è la stampa del risultato e possono partecipare alla competizione quattro giocatori contemporaneamente.

La **First Row** aveva alcune interessanti novità software, tra cui *The Twilight Zone*, una buona avventura grafica per l'Amiga e per il C-64: il giocatore si sveglia nel proprio letto e si accorge di essere stato trasportato... ai confini della realtà. Sempre nelle due versioni troviamo *The Honeymooners*, un'avventura animata con grafica digitalizzata nella quale il giocatore deve raccogliere il denaro sufficiente per partecipare a un meeting (a nessuno viene in mente *The Three Stooges*?). Destinato solo all'Amiga è *Prime Time*, un'ottima simulazione in cui l'utente recita la parte di un impresario che deve sconfiggere la concorrenza vendendo, comprando e creando spettacoli televisivi; da segnalare per la buona interattività. Solo per il C-64 è invece *Moses*, un programma destinato ai bambini, che insegna la storia di Mosè seguendo gli schemi classici di un gioco d'avventura.

La **Gold Disk** era presente con *Movie Setter*, un eccezionale programma per animare disegni IFF dal costo incredibilmente basso (39,95 dollari), che può essere utilizzato anche (sia pur con alcune limitazioni) con soli 512K di memoria. Con lo stesso programma si possono produrre effetti sonori stereofonici e sono già annunciati dischi contenenti ClipSound e ClipArt. Segnaliamo poi *Professional Draw* (200 dollari), un programma grafico simile al famoso *Adobe Illustrator* per il Macintosh: supporta qualsiasi stampante, comprese le laser a colori, ed è adatto anche per disegni tecnici e impaginazione.

Sempre della Gold Disk, è da citare un'interfaccia hardware/software per lo scanner a 300 dpi della Canon: è in grado di "leggere" fogli con grafica e testi e trasferirli nel computer con grande precisione (1.100 dollari).

La **Great Valley Products** (GVP) ha proposto Impact, una sorta di "scatola magica" per l'Amiga 500 che contiene un'espansione da 2 MB (opzionale) e un hard disk da 20, 30, 45 MB o da 40, 80 MB di tipo Quantum (superveloce e di maggior durata). Gli hard disk prevedono l'autoboot con il *Kickstart 1.3*. Impact è una periferica ideale per chi vuole utilizzare professionalmente l'A500 senza passare al 2000.

La **Incognito Software** ha messo in vendita a 79 dollari gli XSpec 3D, occhiali che rendono tridimensionali le immagini sullo schermo. L'effetto però funziona soltanto per i programmi dedicati, e non si nota se non ci si mette a una precisa distanza dallo schermo. Sul versante software, per C-64 e Amiga c'è *Targis*, un gioco a piattaforme piuttosto ovvio e modesto anche come grafica; *Snake Pit*, il solito gioco del serpentone che deve muoversi in un labirinto senza bloccarsi con la sua stessa coda, e *Kingdoms of England*, discreto gioco con vista dall'alto in cui si recita la parte di un cavaliere che deve eliminare i rivali. Solo per l'Amiga, la Incognito Software offre *Atredes BBS 1.1*, un buon programma per la gestione del modem che permette tra l'altro d'inviare file grafici: ha senz'altro alcune qualità, ma risulta anche piuttosto limitato perché permette di scambiare i propri dati soltanto con altri Amiga. Segnaliamo infine *Optics*, un buon programma di ray tracing (simulazione d'ambiente) utilizzabile in multitasking.

La **Irwin Magnetics** è un'azienda leader nel campo degli strumenti di backup (cassette e hard disk) e inoltre costruisce i disk drive per case come Zenith,

Compaq, IBM, HP, NEC, Epson (detiene il 75 per cento del mercato dei disk drive). Ha costruito ora un Tape Retrieve per il Mac, e poiché il Mac utilizza un'interfaccia SCSI come l'Amiga, il Tape Retrieve sarà disponibile anche per l'A2000 non appena verrà prodotto il software per gestirlo.

La **Michtron** prevede di mettere in commercio entro l'inizio dell'89 un sistema a disco laser per l'Amiga, in parte hardware (lettore del disco laser e interfaccia) e in parte software (per la gestione del lettore). Per dimostrarne il funzionamento, veniva proposto *Dragon's Lair* (quello originale). Il software per il disco laser verrà prodotto dalla VIVA (Visual Interfaced Video Authoring). Il prezzo del sistema dovrebbe aggirarsi intorno ai 1.800 dollari, mentre quello di ogni programma su disco laser intorno ai 600.

La **Microprose** sta realizzando una versione per l'Amiga di *Gunship* e *Pirates*: l'uscita è prevista non prima del febbraio '89. Intanto ha realizzato per il C-64 *Red Storm Rising* su disco e cassetta; in pratica si tratta della simulazione di un sottomarino, impegnato nell'avventura tratta dall'omonimo libro. La grafica è pregevole.

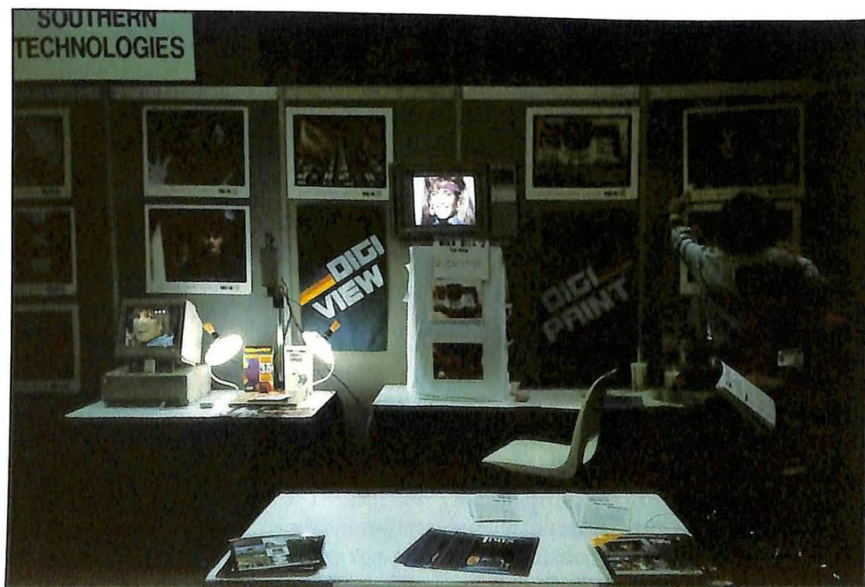
Tra i prodotti presentati dalla **New Horizons**, in evidenza la versione 2.0 di *ProWrite* (uscita nel febbraio 1988), che comunque si limita a sostituire i driver per le stampanti in modo che funzionino anche col *Workbench 1.3*. Inoltre *ProScript*, (uscito nel settembre 1988) che permette di stampare i file di *ProWrite* in formato Postscript (cioè con stampante laser).

Immagini e suoni digitali

La **NewTek**, accanto alla nuova versione 1.3 di *Digi View* (che supporta l'overscan e l'half-brite), ha presentato *Digi Droid*, un motorino che gira automaticamente la piastra di plastica dei filtri del colore, permettendo un certo risparmio di tempo. Per gennaio ha annunciato inoltre *Digi Paint II*, con una bitmap di 1000 per 1000 che permetterà di manipolare un'immagine digitalizzata in overscan muovendola con i tasti cursore. Da citare anche Videotoaster (prezzo intorno ai 1.200-1.500 dollari), un apparecchio annunciato come il più completo digitalizzatore in commercio (dovrebbe servire contemporaneamente da digitalizzatore, frame-grabber e genlock).

La **Precision** ha proposto *Pro Sound Designer*, un programma che funziona con ogni digitalizzatore audio, offrendo l'eliminazione del rumore di fondo, la possibilità di ottenere effetti stereo, di analizzare e modificare un suono e così via. La stessa software house ha presentato anche *Superbase Professional 3*, che a differenza della versione precedente può utilizzare le clipboard e i dati di qualsiasi altro programma.

La **Progressive Peripherals & Software** ha annunciato la disponibilità di un frame grabber in versione PAL in grado di digitalizzare un'immagine in 1/30 di secondo, ottenendo risultati molto nitidi e puliti.

Presente alla mostra anche la **ReadySoft**, con due dei programmi più interessanti visti ultimamente. Il primo è la versione per i dischi magnetici dell'Amiga di *Dragon's Lair*, il famoso arcade su

disco laser (59,95 dollari, sei dischi): il gioco è diviso in due parti, una sezione "labirinto" piuttosto bruttina e una sezione "avventura" infinitamente superiore, che con la sua incredibile grafica a tutto schermo permette di segnalarlo senza esitazioni come il più bel gioco mai apparso per l'Amiga, pur essendo – come l'originale – poco interattivo. Per farlo funzionare sull'Amiga 1000 bastano 512K in quanto elimina il caricamento del *Kickstart*, mentre per l'A500 e per l'A2000 sono necessari 1024K. È disponibile anche la versione PAL. Il secondo programma da segnalare è *Max, the Mac Emulator*, emulatore dell'Apple Macintosh (150 dollari), composto da una parte software e da una parte hardware che richiede l'inserimento nella porta per il secondo drive dell'A500. È comunque necessario inserire una ROM Apple da 64K (50 dollari) o da 128K (150 dollari).

La Readysoft ha proposto anche alcuni giochi di minori ambizioni, come *Cosmic Bouncer* (una pallina che deve rimbalzare su una passerella senza cadere), *Rock Challenge* (un quiz musicale: si deve indovinare chi canta un certo motivo) e *Scary Mutant from Mars* (una classica avventura grafica spaziale con testo e grafica di pessima qualità).

Della **Software Visions** abbiamo potuto vedere alla prova *Microfiche Filer Plus*, un database generale per grafica e testo che collegato a un videodisco permetteva – tramite mouse – di selezionare e vedere qualunque scena del film. Il software costa 179 dollari, ma per realizzare le funzioni descritte è necessaria anche un'interfaccia per videodisco o videoregistratore, che viene venduta a 220 dollari.

La **Spirit Technology** ha proposto un'espansione di memoria Sin 500-2 (160 dollari con 0,5 MB, 299 dollari con 2 MB). La scheda è autoconfigurante e interna, e potrebbe essere per molti una soluzione definitiva per tutti i problemi di memoria dell'A500.

Si è già detto molto dei programmi futuri della **Sublogic**, ma per ora è tutto in fase di progettazione. La realizzazione più vicina nel tempo dovrebbe essere il Scenery disk 9 per *Jet* e *Flight Simulator*, che conterrà Cincinnati, Chicago e zone limitrofe. Al World of Commodore, però, la Sublogic non ha presentato nulla di nuovo, anche se il programma *Jet* per l'Amiga ha continuato a riscuotere molti consensi.

Un annuncio da parte della **Supra Corporation** dovrebbe far rizzare le orecchie degli appassionati di telematica: sarà disponibile entro breve un modem interno per A2000 pienamente Hayes compatibile (179 dollari, con software access junior). Un'altra utile novità è *Printer Buffer* (69,95 dollari) un programma che può

Per ulteriori informazioni contattare direttamente:

Abacus Software
5370, 52nd Street
Grand Rapids, MI 49508, USA
(tel. 001/616-6980330)

Berkeley Softworks
2150 Shattuk Avenue
Berkeley, CA 94704, USA
(tel. 001/415-6440883)

Digitek Inc.
164 W. Seneca, #4
Tampa, FLA 33612, USA
(tel. 001/813-9338023)

First Row Software
900 East 8th Avenue South
King of Prussia, PA 19406, USA
(tel. 001/215-3371500)

Gold Disk Inc.
2171 Dunwin Drive, Unit 13
Mississauga, Ont. L5L1X2, Canada
(tel. 001/416-8280913)

Great Valley Products Inc.
225 Plank Avenue
Paoli, PA 19301, USA
(tel. 001/215-8899411)

Incognito Software
34518 Warren, Suite 149
Westland, MI 48185, USA
(tel. 001/313-4622148)

Irwin Magnetics
1515 West Chester Pike, #1
West Chester, PA 19380, USA
(tel. 001/215-4311592)

Michtron/Microdeal
576 S. Telegraph Road
Pontiac, MI 48053, USA
(tel. 001/313-3345700)

Microprose Software
180 Lakefront Drive
Hunt Valley, MD 21030, USA
(tel. 001/301-7711151)

New Horizons Software
PO Box 43167
Austin, TX 78745, USA
(tel. 001/512-3286650)

NewTek Inc.
115 West Crane Street
Topeka, KA 66603, USA
(tel. 001/913-3541146)

Precision Inc.
8404 Sterling Street, #A
Irving, TX 75063, USA
(tel. 001/214-9294888)

Progressive Peripheral & Software
464 Kalamath Street
Denver, CO 80204, USA
(tel. 001/303-8254144)

Readysoft Inc.
PO Box 1222
Lewiston, NY 14092, USA
(tel. 001/416-7314175)

Software Visions Inc.
26 Forest Road
Framington, MA 01701, USA
(tel. 001/617-8751238)

Spirit Technology Corp.
220 West, 2950 South
Salt Lake City, UT 84115, USA
(tel. 001/801-4854233)

Sublogic Corporation
713 Edgebrook Drive
Champaign, IL 61820, USA
(tel. 001/217-3598482)

Supra Corporation
1133 Commercial Way
Albany, OR 97321, USA
(tel. 001/503-9679075)

The Guitar Connection
6 Wildwood Road
East Chester, NY 10709, USA
(tel. 001/914-7795581)

essere utilizzato per immagazzinare fino a 64K di dati da mandare alla stampante, e poter essere così liberi d'impiegare subito il computer per altri compiti.

Per finire, la **Guitar Connection** ha proposto *Chord Printer*, un pessimo programma per C-64 che dovrebbe insegnare a suonare la chitarra. Spiega di volta in volta su quali corde si devono poggiare le dita, e alla fine di ogni lezione si può ottenere la stampa di tutte le schermate relative.

Le manifestazioni di contorno

Almeno un cenno alla serie di conferenze e seminari organizzati nel corso del World of Commodore è doveroso farlo, anche per dare un'idea di che cosa sono realmente questi show americani.

Tra gli argomenti di cui si è discusso, la grafica sull'Amiga viene al primo posto, ma hanno avuto il loro spazio anche le applicazioni didattiche, la creazione di musica a livello professionale, il desktop publishing, i sistemi di backup, la telematica e l'animazione. Alcuni incontri sono stati dedicati esclusivamente alle domande degli utenti: gruppi di programmatori e di esperti della Commodore erano a disposizione per risolvere i problemi e le curiosità degli intervenuti, tanto per il C-64/128 quanto per l'Amiga.

Non ci sono state troppe celebrazioni per il trentennale della Commodore (ricordiamo che la CBM – Commodore Business Machines – fu fondata a Toronto nel 1958 da Jack Tramiel), ma questa manifestazione, che è la prima completamente sponsorizzata dalla Commodore e dedicata soltanto ai prodotti Commodore, tutto sommato rappresenta la miglior celebrazione a cui si poteva pensare. Adesso si riparte per l'anno 31 aspettando, soprattutto qui in Italia, che tutte le novità per l'89 presentate al World of Commodore arrivino anche nei nostri negozi. ■

GUERRA COMPUTER

Via Bissuola 20/A
Mestre Venezia
Tel. 041/974944 r.a.

Filiali:
Via Vizzotto, 29 S. Donà (Ve)
Via Cairoli, 95 Treviso

Prodotto		Prezzo
Commodore Amiga 500		telefonare
Commodore Amiga 200		telefonare
Commodore C-64 new		telefonare
Commodore C-128 D		telefonare
Commodore PC 10 III		telefonare
Commodore PC 20 III		telefonare
Espansione RAM 256K C-64		telefonare
Espansione RAM 512K A500		telefonare
Espansione RAM 512K A1000		telefonare
Espansione RAM 1MB A1000		telefonare
Espansione RAM 2MB A2000		telefonare
Espansione RAM 8MB A2000		telefonare
Drive Commodore 1541		telefonare
Drive Commodore 1541 II		telefonare
Drive Commodore A1010 per Amiga		telefonare
Drive Commodore A2010 per A2000		telefonare
Drive Compatibile per C-64		telefonare
Drive Compatibile per Amiga	L.	270.000
Sidecar per A1000		telefonare
Dischi Bulk 5 1/4" 2S2D cert.	L.	950
Dischi Bulk 3 1/2" 2S2D cert.	L.	2.200
Framegrabber digitale tempo reale	L.	590.000
Digi view A500/1000/2000	L.	130.000
Digi sound A500/1000/2000	L.	130.000
Niki cartridge II per C-64	L.	65.000
Go Amiga cartridge per C-64	L.	65.000
Tappetino salva mouse	L.	18.000
Stampante Commodore MPS 1500	L.	570.000
Stampante Commodore MPS 1250	L.	490.000
Interfaccia Midi per A500/1000/2000	L.	49.000
Emulatore C-64 A500/1000/2000	L.	49.000
Scheda Janus XT		telefonare
Scheda Janus AT		telefonare
Monitor Commodore 1084		telefonare
Monitor Commodore 2080		telefonare
Hard disk SCSI per A2000		telefonare
Hard disk PC/Amiga per A2000		telefonare
Scheda video RF per A2000		telefonare
Genlock videomaster per A500		telefonare

Offerte del mese:

Adattatore telematico 6499 con abbonamento gratuito per un anno per il collegamento al Videotel e manuale di istruzione

L. 39.000

Cofanetto programmi originali Amiga con manuale in italiano tra cui: Superbase, Logistix, Music studio, Super Huey, grafica pittorica e Cad, gestione archivi, recupero dischi danneggiati, emulatore di terminali e trasmissione dati, laboratorio musicale, giochi vari.
I programmi sono corredati di manuale in italiano e garantiti dalla C.T.O. Italia

L. 125.000

Stampante Panasonic KX 1081 Centronics 160 CPS NLQ Draft ottima per Amiga e personal computer

L. 430.000

Siamo inoltre distributori di prodotti: Commodore, Philips, Amstrad, Atari, Star, Nec, Citizen, Panasonic, Seikosha, Olivetti Prodest. Abbiamo una grande quantità di software per Amiga, C-64, MS-DOS, Atari ST, MSX 1 e 2: scriveteci o telefonateci, in quanto non possiamo elencarvi tutto ciò che abbiamo in catalogo. I prezzi sopra elencati sono IVA compresa e al netto delle spese di spedizione. La merce è garantita per un anno dalla data d'acquisto. Cercheremo di soddisfare ogni vostra esigenza. Abbiamo la più vasta scelta di hardware e software del triveneto.

LE STAMPANTI FUJITSU DX2300 E DL3300

Due stampanti che propongono intelligenti soluzioni tecniche e un'estrema cura dei particolari a un prezzo adeguato alle prestazioni. Velocità, qualità di stampa, programmabilità e un kit colore di alto livello

di Luca Giachino

Affidabili, rapide, silenziose, esteticamente riuscite, ricche di opzioni per la stampa, curate nei particolari e pensate con un occhio sempre attento alle piccole e grandi esigenze dell'ufficio... stiamo parlando delle DX2300 e DL3300, due stampanti destinate a impieghi professionali che vengono proposte dalla Fujitsu a un prezzo sicuramente adeguato alle loro qualità. Con la sua linea di stampanti, la Fujitsu sostiene una filosofia semplice e rara: qualità professionale per lavorare meglio e prezzi competitivi per valorizzarla.

Ma entriamo subito nel vivo della nostra prova. I due modelli che sottoponiamo all'attenzione dei lettori hanno evidentemente la stessa matrice costruttiva: per quanto le loro prestazioni e le loro caratteristiche tecniche siano diverse, la cura che caratterizza il modello a 24 aghi è la stessa che si riscontra nella DX2300, a 9 aghi. Si tratta di periferiche pronte a soddisfare una pluralità di esigenze, dalla stampa rapidissima di grandi quantità di dati alla produzione di stampati più eleganti e raffinati (eventualmente anche a colori).

Le considerazioni che emergeranno nel corso della prova valgono anche per i modelli DX2400 e DL3400, la cui unica differenza rispetto ai modelli in prova è la maggiore larghezza del foglio di stampa. La serie 300 produce linee di stampa la cui larghezza massima è 10 pollici (stampando a 10 cpi si ottengono i consueti 80 cpl, caratteri per linea), mentre la serie 400 produce linee di stampa la cui larghezza massima è 15 pollici (stampando a 10 cpi si ottengono 136 cpl).

Iniziamo puntando i riflettori sulla DX2300, il modello di punta della serie Fujitsu a 9 aghi, che ci ha favorevolmente sorpresi fin dall'inizio, dimostrando prestazioni superiori a quelle di altre macchine collocabili nella stessa fascia di prezzo.

La DX2300

La prima impressione di fronte alla DX2300 è di un'elevata professionalità. Il disegno è piacevole, e i toni freddi dei colori (ghiaccio per il cabinet e grigio fumé per le parti semitrasparenti) sono perfetti per individuare subito la macchina come appartenente a una categoria superiore. Maneggiandola, poi, se ne rileva subito la robustezza: la struttura in materiale plastico è solida e non si deforma facilmente. Sul lato superiore risiede la console operativa, che oltre alle normali funzioni di controllo dell'avanzamento carta e dello stato della stampante, permette di attivare la stampa NLQ semplicemente premendo un tasto, e di configurare alcuni parametri di stampa senza estrarre il foglio inserito e senza dover accedere ai DIP switch. I tasti di controllo sono protetti da uno strato di plastica morbida che copre uniformemente tutta la console, e dovrebbe rendere più difficili eventuali infiltrazioni di polvere.

Sempre sul lato superiore risiede lo sportello semitrasparente fumé che protegge la meccanica di trascinamento. Questo sportello è dotato di due snodi, che gli permettono di coprire una parte del foglio di stampa quando è montato il sostegno guida per l'inserimento verticale dei fogli,

riducendo ulteriormente il rumore di stampa, già ampiamente limitato dal materiale fonoassorbente con cui è rivestito l'interno della stampante. Tutta questa cura nel ridurre il rumore il più possibile dà ottimi risultati: la stampa è davvero silenziosa, con grande soddisfazione dei nostri nervi già afflitti da decine e decine di altre macchine ad alto quoziente di rumorosità.

Lo sportello è vincolato a un altro, più grande, che costituisce parte integrante del cabinet. Aprendo quest'ultimo si accede al sistema di trascinamento della testina, operazione da effettuare quando si deve sostituire il nastro esaurito o montare lo speciale kit colore opzionale, tramite il quale si può trasformare la DX2300 in una stampante a colori.

Sulla destra dello sportello risiede una levetta a slitta per la commutazione fra il trascinamento a frizione e quello a trattore. Dietro si trova un altro sportello, estraibile, che permette di accedere alla scheda dell'interfaccia e alla scheda che contiene il firmware di gestione (si può far installare a scelta l'interfaccia seriale RS-232 o l'interfaccia parallela Centronics). Per estrarre lo sportello è sufficiente svitare con le dita una grossa vite sul retro della stampante, operazione che permette di accedere anche ai DIP switch sulla scheda del firmware. Sulla sinistra, infine, risiede l'interruttore d'accensione. È interessante notare che anche l'interruttore è stato disegnato con una certa attenzione all'armonia generale, a riprova di una cura che va dai materiali impiegati all'aspetto estetico dell'intero prodotto.

Sul retro, infine, oltre al connettore dell'interfaccia e alla vite che serra lo sportello, si trovano la presa per il cavo d'alimentazione, il copri-fusibile, e due fori nel telaio metallico interno ai quali si possono collegare due fili per la massa, sempre utili quando si connettono periferiche alimentate separatamente. Nell'insieme, la DX2300 appare quindi funzionale ma anche di elegante design: nessuno sarà costretto a nasconderla perché non s'intona con il suo arredamento. Anche a una prima analisi, risulta evidente che la Fujitsu ha cercato di curare le sue macchine veramente in ogni dettaglio.

I modi di stampa

La DX2300 dispone soltanto di una testina a nove aghi, ma riesce a ottenere una velocità di stampa notevolissima anche senza sacrificare la qualità, soprattutto in NLQ. In modo draft arriva fino a 324 cps (caratteri per secondo), con una densità di stampa di 12 cpi (caratteri per pollice), pari a 96 cpl (con la DX2400 a 12 cpi si ottengono invece 163 cpl).

Si tratta di una velocità che compete con quella delle stampanti di maggior livello, ed è un dato ancor più interessante se diamo un'occhiata al prezzo. Con una velocità di stampa così elevata, la DX2300 diventa particolarmente idonea a servire code di stampa provenienti da più computer in rete, senza che ogni ufficio debba attendere troppo a lungo. E per di più i caratteri stampati sono più che leggibili e non sembrano risentire dell'elevata velocità: so-

no ben marcati e mantengono bene l'allineamento verticale. Merito sicuramente di una meccanica progettata con competenza e perizia.

Ma non facciamoci abbagliare dai record, perché – se escludiamo i clienti che hanno esigenze limitate a una sola prestazione particolare – il modo migliore per valutare una stampante è tener conto del complesso delle sue caratteristiche. Continuiamo quindi con i dati numerici. In modo draft, la DX2300 riesce a stampare 270 cps a 10 cpi, e 231 cps a 17,1 cpi (per chi non lo sapesse, diciamo che la media, per stampanti dello stesso tipo si attesta in genere attorno ai 200 cps). In modo NLQ la velocità si riduce drasticamente a 54 cps con 10 cpi, un risultato comunque più che accettabile.

Anche se la rapidità di stampa ha generalmente un peso rilevante nella valutazione di una stampante, la qualità del carattere non è da meno. La DX2300 mantiene un'elevata qualità a tutte le velocità.

Mentre nella stampa draft il carattere è composto da una matrice di 9 x 7 punti, nella stampa NLQ la matrice diventa di 19 x 16 punti. Inoltre, in NLQ la DX2300 impiega una fonte carattere molto elegante. Se si desidera aver la possibilità di dare ai propri stampati un aspetto estetico di alto livello, questa stampante è sicuramente all'altezza della situazione (tenendo sempre conto dei limiti intrinseci della stampa ad aghi).

La DX2300 è in grado di emulare la Epson FX-80, la IBM Graphics Printer, e la IBM Proprinter. A seconda del tipo di emulazione, le prestazioni di stampa variano leggermente. In emulazione Epson, la DX2300 offre la maggior varietà di opzioni. La densità di stampa è selezionabile fra 10, 12, 15, 17,1 e 20 cpi,

Fujitsu DX2300/2400:
prova di stampa

Qualita' draft

Tondo:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Corsivo:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Sottolineato:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Nero:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Doppio passaggio:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Stampa espansa:
ABCDEFGHIJKLM
abcdefghijklm

Stampa a 12 cpi:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Stampa a 17,1 cpi:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Qualita' NLQ

Tondo:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Corsivo:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Sottolineato:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Stampa espansa:
ABCDEFGHIJKLM
abcdefghijklm

Stampa a 12 cpi:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Stampa proporzionale:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

e si può attivare anche la spaziatura proporzionale. Gli stili di stampa sono i consueti tondo, nero, corsivo, sottolineato, apice, pedice, e per marcare ulteriormente i caratteri è possibile attivare il doppio passaggio della testina, o il modo di stampa enfatizzato. Qualora sia necessario stampare caratteri molto particolari, è possibile caricare nella memoria RAM della stampante un intero set di caratteri, contenente sia caratteri draft che NLQ. Infine è possibile abilitare il modo di stampa esadecimale; si tratta di una caratteristica utile quando si ha l'impressione che l'interfacciamento software fra il computer e la stampante non sia dei migliori, e si renda quindi necessario verificare esattamente quali caratteri giungono alla stampante.

Le caratteristiche

Per variare la sua configurazione permanente occorre accedere ai DIP switch installati sulla scheda del firmware. Se invece l'intervento ha carattere temporaneo, è possibile attivare alcune opzioni tramite la console operativa, oltre che via software. Il programma che gestisce queste opzioni si avvale di un beep per comunicare con l'utente, e non richiede quindi l'uso di un foglio di stampa ma, dal momento che non segnala lo stato attuale di ogni parametro, non risulta un sistema particolarmente comodo; comunque esiste la possibilità di riportare tutti i parametri al loro stato di default, per tenere sempre sotto controllo la situazione. Fra le opzioni che si possono attivare da console le più importanti sono la stampa compressa, espansa, elite e monodirezionale, la lunghezza del foglio e il salto della perforazione. Sono tutte opzioni che si possono attivare rapidamente e senza estrarre il foglio di stampa. Si tratta di un sistema utilissimo, anche se un maggior ventaglio di possibilità e un programma di gestione più sofisticato lo avrebbe messo maggiormente in evidenza.

Fra le varie caratteristiche di questa stampante, è interessante la possibilità di mantenere nel meccanismo di trascinamento il modulo continuo mentre si attiva la frizione per stampare su un foglio singolo. La DX2300 consente infatti di spostare all'indietro il modulo continuo fino a quando non esce dal rullo, lasciandolo perfettamente in presa nei trattori. Per effettuare questa operazione, è sufficiente premere il tasto di form feed quando la stampante è on-line. A ogni pressione il modulo continuo retrocede di un foglio, e quindi l'utente non deve far altro che continuare a premere il tasto fino a quando il modulo è completamente uscito dal rullo. A questo punto, attivando la frizione tramite l'apposita levetta, si procede alla stampa sui fogli singoli. Quando non si ha più bisogno della frizione, si riattivano i trattori, e con una serie di form feed a stampante off-line si fa avanzare la carta fino al foglio desiderato: è una possibilità che risulterà graditissima a chi ha l'abitudine di scambiare spesso i due sistemi di trascinamento, perché estrarre il modulo continuo – oltre a essere fastidioso – comporta sempre il rischio di sporcarlo o spiegazzarlo.

La DX2300 dispone di un sostegno orientabile per l'inserimento verticale dei fogli singoli. Prevede inoltre la collocazione automatica del foglio in posizione di stampa, rendendo ancora più agevole ogni operazione (opzionalmente, è possibile acquistare il caricatore automatico dei fogli singoli).

Tornando alla configurazione, salta all'occhio la possibilità di selezionare, tramite i DIP switch, otto diverse lunghezze del foglio di stampa, che permettono di stampare su fatture, etichette, buste, e qualsiasi altro tipo di modulo dotato di lunghezza inconsueta. Addentrandoci nell'esame, notiamo sempre di più che la DX2300 è una stampante le cui caratteristiche, da quelle macroscopiche ai dettagli, sono tutte tese a offrire la versatilità e la professionalità richieste da un impiego professionale.

Il buffer di stampa di serie è da 10K, ma è espandibile a 18K. La DX2400, invece, dispone di un buffer di serie da 8K, espandibile a 16K. Anche con i buffer di serie, le DX permettono al computer collegato di concludere la comunicazione con circa cinque pagine di anticipo. Se il computer non è multitasking, e non prevede lo spool (simultaneous peripheral operations on-line) di stampa in background, la discreta capienza del buffer e la notevole velocità di stampa permettono comunque di snellire il processo.

Il manuale, per quanto in lingua inglese, è molto ben strutturato e comprensibile. Abbonda di figure e di dati tecnici, e descrive ogni aspetto hardware/software che è necessario conoscere per un corretto impiego della DX2300.

Il colore

Acquistando separatamente un apposito kit, è possibile dotare la DX2300 della stampa a colori. Considerando che il costo del kit rappresenta un sesto del valore commerciale della stampante, si tratta senz'altro di un optional da prendere in considerazione, soprattutto per chi possiede un Amiga.

Il kit non è altro che un semplice congegno meccanico da fissare tramite due viti sul carrello che trascina testina e nastro, e un nastro inchiostrato contenente oltre al nero i tre colori fondamentali: giallo, rosso e blu. A chi lo acquista, la Fujitsu fornisce anche nella confezione un foglietto con le istruzioni di montaggio, e un cacciavite. Ancora una volta sono da sottolineare gli sforzi della casa produttrice per mettere il cliente a proprio agio. Il congegno meccanico è dotato di un piccolo motorino che quando riceve corrente produce la rotazione di una spirale di plastica. In questa spirale è incastrata una levetta che a seconda del verso di rotazione del motorino si alza e si abbassa con

un'escursione di circa 2 cm, quanto basta per sollevare e abbassare il nastro in modo da selezionare il giusto colore da imprimere di volta in volta sulla carta.

L'installazione del kit non è semplice come potrebbe sembrare a prima vista, ma prestando la dovuta attenzione non è necessario perdere più di un quarto d'ora. Possiamo anticipare che anche nella stampa a colori la velocità è elevata, a ulteriore conferma dell'efficienza di questa stampante.

La prova e l'analisi interna

Abbiamo eseguito la prova collegando la DX2300 a un Amiga. Ovviamente avremmo potuto collegarla anche a un C-64, ma è con l'Amiga che le caratteristiche grafiche vengono davvero sfruttate, e inoltre l'accoppiata C-64/DX2300 ci sembra alquanto improbabile.

La stampante ha dimostrato, come già abbiamo in parte anticipato, un'ottima affidabilità e una qualità di tutto rispetto. L'attenzione richiesta all'utente per svolgere qualsiasi operazione è ridotta al minimo indispensabile: la collocazione automatica dei fogli in posizione di stampa, e la possibilità di lasciare sempre in presa il modulo continuo sono caratteristiche che potrebbero anche apparire poco interessanti, ma non bisogna dimenticare che se una periferica viene usata in modo intensivo e per lungo tempo, anche le mancanze e i difetti più irrilevanti vengono al pettine e diventano motivi di disagio.

I tasti della console operativa si utilizzano senza difficoltà, e l'inserimento dei fogli singoli, con sostegno e guide verticali, non crea alcun problema. Ci è sembrato degno di nota il facile accesso ai DIP switch sulla scheda del firmware, e soprattutto la possibilità di modificarli dall'alto, senza bisogno di spostare la stampante per accedere a qualche punto nascosto. Comunque il loro uso non è certo frequente, visto che le funzioni di base sono attivabili da console.

Con l'Amiga abbiamo impiegato il driver di stampa Epson JX-80. Nella prova abbiamo cercato di portare la DX2300 ai suoi limiti estremi in molti modi diversi. Le tre aspettative principali si sono pienamente verificate: ottima la velocità di stampa, che non compromette l'affidabilità della meccanica, buona la qualità (soprattutto in NLQ dal momento che la DX2300 utilizza una fonte carattere molto più elaborata di quella impiegata in modo draft), soddisfacente la bassa rumorosità a tutte le velocità.

Dalla stampa dei testi, si passa poi a quella "grafica", a colori, dove la DX2300 ci ha riservato piacevoli sorprese. Innanzitutto, non si è verificato il rallentamento che ci aspettavamo: grazie all'ottimizzazione dei percorsi e all'elevata velocità di spostamento del carrello, la stampa a colori si è rivelata sorprendentemente rapida. E questo senza compromettere assolutamente la qualità, che si è rivelata del tutto accettabile. È

inutile ribadire ancora i limiti che la tecnologia di stampa ad aghi incontra nell'uso dei colori e in genere nella grafica, ma dobbiamo sottolineare un aspetto della DX2300 che ci ha favorevolmente impressionati: la consueta striatura che spesso si forma fra le righe di stampa, dovuta principalmente all'inchiostro che sul nastro non è perfettamente uniforme e a problemi di allineamento meccanico durante l'avanzamento, con la DX2300 è del tutto inesistente. Abbiamo perfino stampato una schermata monocolore, e l'uniformità ottenuta ci ha sorpresi. Il kit colore si è rivelato uno strumento hardware tanto semplice quanto efficace, e non è certo una cattiva idea renderlo un acquisto opzionale, anche se spesso gli acquirenti assumono un atteggiamento diffidente nei confronti di una dotazione non standard. Ma il riscontro pratico ha fugato ogni dubbio, facendoci apprezzare ancor di più la professionalità della Fujitsu. In pratica, con un aumento di spesa di circa il 15 per cento e un po' di fai da te, la DX2300 diventa anche un'ottima stampante a colori.

Ancora qualche considerazione per concludere la nostra analisi: la sostituzione del caricatore d'inchiostro è davvero agevole grazie all'intelligente sistema d'aggancio adottato; la sede del caricatore e la forma della testina permettono di eseguire l'operazione semplicemente appoggiando il caricatore ed esercitando una leggera pressione.

Per aprire la DX2300 è sufficiente svitare quattro viti e sollevare la parte superiore del cabinet avendo cura di disconnettere la console operativa. La prima cosa che salta agli occhi è l'ordinata organizzazione interna: osservando le parti meccaniche "dietro le quinte", è fonte di soddisfazione constatare che anche il cablaggio interno è curato, e non si è lesinato né sui materiali metallici, né sulla quantità di materiale fonoassorbente. Tutti i collegamenti fra i vari circuiti sono effettuati tramite connettori che ne facilitano la rimozione.

La scheda di controllo della stampante è estremamente sofisticata. Su di essa risiedono la

CPU che controlla l'intera stampante, e gli integrati di I/O per l'interfacciamento con l'hardware. Due connettori inoltre accolgono la scheda dell'interfaccia (seriale o parallela) e la scheda del firmware. Di quest'ultima (vedere foto sopra) in particolare va segnalata la notevole comodità d'estrazione. Basta infatti sollevare due alette snodate per estrarla senza correre il rischio di comprometterne l'integrità. Una delle due alette protegge inoltre i DIP switch da manomissioni involontarie. Si può estrarre la scheda di controllo e l'interfaccia semplicemente aprendo lo sportello scorrevole, senza bisogno di aprire l'intera macchina.

La possibilità di sostituire molto facilmente la scheda del firmware lascia supporre che la Fujitsu terrà aggiornato il software di gestione della stampante. Con ogni probabilità, se si affermeranno nuovi standard l'utente non dovrà fare altro che sostituire una scheda. Si tratta di un'indubbia comodità.

Conclusioni

Difficile non restarne conquistati. La DX2300 offre prestazioni di alto livello: elevata velocità e qualità di stampa, comodità d'uso e affidabilità; tre prerogative di sicuro successo.

Le nostre conclusioni sono senza dubbio estremamente positive, soprattutto perché in ogni sezione della prova (il manuale, l'hardware, i risultati ottenuti su carta, l'affidabilità...) ci siamo imbattuti in un'estrema cura dei particolari, anche di quelli meno importanti.

A un costo certo non invitante per il possessore di un home computer, ma proporzionato alle sue caratteristiche da stampante professionale, la DX2300 è in grado di offrire una qualità che va ben oltre le normali aspettative e risponde perfettamente alle esigenze dei settori professionali. E con il kit colore, diventa un prodotto molto interessante anche per i più esigenti utenti dell'Amiga.

DATI TECNICI

	DX2300	DX2400
Testina di stampa	9 aghi a impatto	
Velocità di stampa	324 cps draft a 12 cpi 270 cps draft a 10 cpi 231 cps draft a 17,1 cpi 54 cps NLQ a 10 cpi	
Densità di stampa	10/12/15/17,1/20 cpi e proporzionale	
Caratteri per linea	80 cpl 96 cpl 120 cpl 137 cpl 160 cpl	136 cpl 163 cpl 204 cpl 233 cpl 272 cpl
Buffer di stampa	2, 10, 18K	8, 16K
Trascinamento	A trattore per moduli perforati A frizione per fogli singoli	
Emulazioni	IBM Graphics Printer IBM Proprinter Epson FX-80/JX-80	
Interfaccia di serie	Parallela Centronics o seriale	
Dimensioni	120 x 438 x 345 mm	120 x 580 x 345 mm
Peso	9,2 kg	11,7 kg
Prezzo al pubblico	L. 1.250.000 + IVA (p) L. 1.310.000 + IVA (s) (p - parallela, s - seriale)	L. 1.600.000 + IVA (p) L. 1.660.000 + IVA (s)
	Optional	
Caricatore automatico di fogli singoli	L. 520.000 + IVA	L. 580.000 + IVA
Kit colore	L. 210.000 + IVA	L. 210.000 + IVA

Importatore

Fujitsu Italia Spa
Via Melchiorre Gioia, 8 - 20124 Milano (tel. 02/6572741/2/3/4/5)

La DL3300

La DL3300 è una stampante che per alcuni aspetti somiglia alla DX2300. Mantiene infatti lo stesso disegno per il cabinet, i colori, la robustezza, la silenziosità, e gli stessi pregi, come la qualità di stampa anche ad alta velocità e l'affidabilità. Si tratta però di un modello superiore, che aggiunge soluzioni tecnologiche e funzionali di estremo rilievo.

Anzitutto, la DL3300 è dotata di una testina di stampa da 24 aghi, grazie alla quale si ottengono caratteri ad alta densità di punti, che oltre ad apparire molto ben definiti sono continui anche nelle linee più sottili.

La DL3300 offre una velocità massima di stampa di 288 cps, cinque fonti carattere residenti e altre sei immagazzinate in tre cartucce d'espansione che si possono acquistare a parte. Inoltre è disponibile una grande varietà di stili, spaziature, altezze e larghezze dei caratteri con soluzioni particolari per il word processing; differenziazione fra spaziatura e dimensione del carattere, controllo *totale* da console operativa sia dei parametri permanenti (con conseguente eliminazione dei DIP switch) sia dei parametri provvisori; possibilità di creare due configurazioni permanenti dei parametri richiamabili da console; caricatore opzionale per fogli singoli con uno o due cassetti (con due cassetti è possibile mantenere sempre pronti fogli di due formati diversi). Si tratta di opzioni che rendono la DL3300 in grado di soddisfare le esigenze più disparate a un livello molto alto. Si può stampare una lettera con i caratteri quadruplicati in altezza, per esempio, oppure stampare un comunicato dove i caratteri – nei punti fondamentali – mantengono il corpo del testo, ma sono molto spaziati fra loro. Per queste operazioni, e molte altre, è sufficiente intervenire opportunamente da console.

Sono comunque tutte caratteristiche che vanno ad aggiungersi a quelle che abbiamo già descritto nella prima parte dell'articolo. La possibilità di mantenere in presa il modulo continuo quando agisce la frizione per i fogli singoli, il meccanismo di collocazione automatica della carta in posizione di stampa, il sostegno e le guide verticali per i fogli singoli e così via, sono tutte opzioni su cui non torneremo, ma che appartengono tanto alla DX2300 quanto alla DL3300.

Anche esternamente le due macchine sono molto simili, quindi mettiamo solo in evidenza le poche differenze: la console operativa della DL3300 è molto più elaborata, e può svolgere un maggior numero di funzioni. Con stupefacente semplicità è possibile scegliere fra quattro diverse fonti di stampa e due configurazioni complete di parametri, che restano in memoria anche quando la stampante viene spenta.

Anche la DL3300 dispone di uno sportello estraibile che cela la scheda d'interfaccia (seriale o parallela). A differenza della DX2300, il firmware di gestione è stato conglobato sulla stessa scheda dell'interfaccia; dove la DX2300 contiene la scheda del firmware, la DL3300 possiede una struttura in plastica destinata ad accogliere le cartucce per le fonti carattere addizionali. Per inserire

una delle tre cartucce opzionali non è comunque necessario rimuovere lo sportello. A questo scopo è stata prevista una feritoia e un pulsante d'estrazione della cartuccia. Non essendo dunque necessario aprirlo in nessun caso (visto che mancano anche i DIP switch), lo sportello questa volta è a incastro ed è senz'altro meno agevole da estrarre. Per il resto, esternamente le due stampanti sono identiche, anche nelle dimensioni.

La velocità e i modi di stampa

Nella serie Fujitsu a 24 aghi, la DL3300 rappresenta il modello inferiore. In termini di velocità di stampa, si discosta molto dai migliori modelli a 24 aghi della stessa famiglia, che arrivano a vette di 486 cps in qualità bozza e 162 cps in qualità lettera, ma si deve sempre tenere in considerazione il suo prezzo relativamente basso per una stampante professionale. Con la fonte meno elaborata, corrispondente alla qualità Hi-Draft, a 10 cpi si ottengono 240 cps, mentre a 12 cpi la velocità diventa 288 cps. Con la fonte di qualità intermedia, la qualità Draft, a 10 cpi si ottengono 180 cps, mentre a 12 cpi la velocità diventa 216 cps. Alla massima qualità di stampa (per la quale la DL3300 mette a disposizione tre fonti carattere residenti), chiamata qualità Letter, a 10 cpi si ottengono 60 cps mentre a 12 cpi la velocità diventa 72 cps. Su questi dati possiamo fare due considerazioni. La DL3300 offre una vasta scelta di fonti carattere, e quindi di velocità di stampa, riuscendo a coprire esigenze molto diverse. In secondo luogo, sebbene la velocità massima non sia particolarmente elevata in confronto ad altre stampanti a 24 aghi, 60 e 72 cps in qualità Letter costituiscono comunque un'ottima prestazione.

La DL3300 mette a disposizione dell'utente cinque fonti carattere residenti: due rispettivamente per la qualità Draft e Hi-Draft, e tre per la qualità Letter. In qualità Letter i caratteri delle tre fonti disponibili si differenziano sia nella forma sia nella larghezza. La fonte Courier contiene i caratteri più larghi (circa otto centesimi di pollice), la fonte Prestige Elite contiene caratteri meno larghi (circa sei centesimi di pollice), e la fonte Compressed contiene i caratteri più stretti in assoluto (circa

---

Fujitsu DL3300/3400: prova di stampa

Fonti carattere residenti:

Courier
abcdefgh IJKLMNOP 01234567

Prestige Elite
abcdefgh IJKLMNOP 01234567

Compressed
abcdefgh IJKLMNOP 01234567

Draft
abcdefgh IJKLMNOP 01234567

Hi-Draft
abcdefgh IJKLMNOP 01234567

Alcuni stili disponibili:

Qualita' Letter:	Qualita' Draft:	Qualita' Hi-Draft:
Sottolineato	Sottolineato	Sottolineato
Corsivo	*Corsivo*	*Corsivo*
Ombreggiato	**Ombreggiato**	**Ombreggiato**
Nero	**Nero**	**Nero**
Apici	Apici	Apici
Pedici	Pedici	Pedici
Compresso	Compresso	Compresso

Dimensioni particolari del carattere:

Caratteri alti il doppio

Caratteri alti quattro volte

---

cinque centesimi di pollice). È importante sottolineare che la larghezza dei caratteri presenti in ogni fonte prescinde dalla spaziatura impostata, e permette quindi di combinare caratteri stretti (fonte Compressed) con spaziature estese (10 cpi), o viceversa caratteri larghi (fonte Courier) con spaziature minime (fino a 20 cpi). Aver reso indipendenti queste grandezze è un piccolo particolare non certo fondamentale nella nostra valutazione, ma che fa capire quanto sia alta la varietà di combinazioni che si possono ottenere.

Le altre due fonti residenti vengono impiegate nella stampa più rapida. Quella utilizzata in modo Hi-Draft è di un'assoluta semplicità ed essenzialità, ma grazie ai 24 aghi della testina è sempre molto leggibile e quindi idonea per veloci riscontri preliminari. In modo Draft il carattere, per quanto risulti meno nitido che in qualità Letter, è ben delineato, e per essere un carattere Draft è più che presentabile.

A tutte queste fonti sono poi applicabili un grande numero di spaziature e stili. La DL3300 offre le seguenti spaziature: 10 cpi (80 cpl, 136 cpl per la DL3400), 12 cpi (96 cpl, 163 cpl per la DL3400), 15 cpi (120 cpl, 204 cpl per la DL3400), 17,1 cpi (136 cpl, 232 cpl per la DL3400), 18 cpi (144 cpl, 244 cpl per la DL3400), 20 cpi (160 cpl, 272 cpl per la DL3400), e proporzionale; nella stampa espansa, cioè con spaziatura superiore a un decimo di pollice, la DL3300 offre le seguenti possibilità: 2,5 cpi (20 cpl, 34 cpl per la DL3400), 3 cpi (24 cpl, 40 cpl per la DL3400), 5 cpi (40 cpl, 68 cpl per la DL3400), 6 cpi (48 cpl, 81 cpl per la DL3400). Come si vede, le spaziature disponibili sono davvero tante, e una tale varietà può rendersi molto utile, per esempio, nell'esatta compilazione di moduli prestampati. Gli stili disponibili sono i consueti corsivo, nero, sottolineato, apice e pedice, espanso, ai quali la DL3300 aggiunge lo stile ombreggiato e condensato (con questo stile, dal momento che il carattere diventa molto compresso, la spaziatura scende automaticamente a 18 cpi; non si confonda questo stile con la fonte Compressed disponibile in qualità Letter).

Per quanto riguarda la compatibilità, la DL3300, come c'era da aspettarsi, è in grado di emulare diverse stampanti. Le emulazioni disponibili sono: le Fujitsu DPL24C, DPL24I e DPL24D, le IBM Graphics Printer, Proprinter XL, le Epson FX-80 e JX-80, la Diablo 630. Nell'ottimo manuale che accompagna questa stampante è perfino riportata una tavola che elenca praticamente tutte le più famose stampanti disponibili sul mercato, e le relative emulazioni che è necessario abilitare. Si tratta del manuale per l'utente. La Fujitsu rende disponibile a richiesta anche il manuale per il programmatore.

Il controllo da console, il modo Setup

La DL3300 dispone di un efficientissimo controllo da console. Si può intervenire sui parametri permanenti della stampante, entrando nel modo Setup, oppure intervenire sulla sua configurazione in modo temporaneo, entrando nel modo Special.

Per abilitare il modo Setup è necessario che sia inserito un modulo continuo, e che mentre la stampante viene accesa si tenga premuto un tasto della console: la DL3300 entra automaticamente nel programma di configurazione dei parametri permanenti, stampando un breve riassunto dei tasti utilizzabili per dialogare con il programma, e il primo menu di voci selezionabili, il menu principale. Agendo con un tasto, l'utente muove la testina finché il cursore rosso stampato sulla plastica trasparente non arriva sotto la voce prescelta del menu. Quando si preme il tasto di conferma, il programma visualizza tutti i parametri relativi a quella voce offrendo all'utente la possibilità di alterarli.

La prima voce del menu principale – Panel – permette di selezionare le alternative tra le varie fonti da utilizzare poi nel modo Special. Si devono impostare tre parametri: Draft per la fonte ad alta velocità (l'alternativa è fra Draft e Hi-Draft), Font1 e Font2 per le fonti in qualità Letter e le spaziature (si può scegliere fra 13 fonti – tre residenti, otto sulle cartucce, e due caricabili nella memoria RAM – e fra 11 spaziature).

Le due voci successive, Menu1 e Menu2, permettono d'impostare due diverse configurazioni permanenti composte da 20 parametri che potranno in seguito essere richiamate da console nel modo Special. I parametri sono gli stessi per Menu1 e Menu2: lo scopo è offrire all'utente due configurazioni attivabili con una semplice combinazione di tasti, senza bisogno d'intervenire sui parametri permanenti. Torneremo sull'argomento nella descrizione del modo Special.

Le altre voci del menu principale non hanno più niente a che fare con il modo Special. Selezionando la voce Hardware si può attivare o disattivare il beep, attivare la stampa monodirezionale e la gestione dei due caricatori automatici di fogli singoli, impostare la lunghezza delle word in arrivo all'interfaccia, stabilire la quantità di memoria RAM per il buffer (256 byte, 2K, 8K o 24K).

Quando si seleziona la voce seguente, List, il programma stampa una tavola di riferimento dei parametri permanenti in due diverse dimensioni (vedere la tavola della pagina successiva). È un'idea abbastanza utile perché permette di avere automaticamente un promemoria dei parametri impostati in Menu1 e Menu2 e richiamabili da console nel modo Special. È consigliabile tenere sempre questo foglio accanto alla stampante.

Seguono poi la voce Default, per far tornare tutti i parametri permanenti e quelli temporanei ai loro valori di default, la voce Self-test per avviare il test di

stampa, la voce Hex-dump per abilitare la stampa esadecimale dei caratteri che giungono all'interfaccia, la voce V-alnmnt per controllare gli allineamenti verticali e quindi la corretta taratura della meccanica di stampa, la voce Tof-adj per impostare lo spazio che la stampante deve lasciare all'inizio e alla fine di un foglio di stampa quando salta la perforazione e predispone la nuova pagina, e infine la voce Save/end per salvare in modo permanente tutte le variazioni introdotte, e uscire dal programma.

Anche da questa presentazione sommaria emerge la complessità del programma e la quantità di parametri su cui è possibile intervenire per personalizzare il comportamento della stampante. Senza spingerci oltre nell'analisi del modo Setup, evidenziamo i parametri più significativi di Menu1 e Menu2.

È possibile raddoppiare o quadruplicare le dimensioni di default dei caratteri, sia in altezza sia in larghezza. Giocando su questi due parametri è possibile, per esempio, stampare in caratteri più grandi una lettera che si deve inviare tramite telefax, o un comunicato da appendere nella bacheca dell'ufficio.

Si può variare l'interlinea fra otto diversi valori (da una riga di stampa a otto righe). Questo parametro diventa importante quando si aumentano le dimensioni verticali dei caratteri, dal momento che manca la selezione automatica dell'interlinea appropriata.

Vi sono ben 14 alternative per la lunghezza del foglio di stampa, per venire incontro alle esigenze più disparate (3,5", 4", 5", 5,5", 6", 7", 8", 8,5", 11", 11,6", 12", 14", 18").

Infine si deve selezionare l'impaginazione (nove alternative) che mantiene sempre margini destro/sinistro simmetrici, la nazionalità dell'alfabeto, l'emulazione, la fonte, lo stile, la spaziatura, il set di caratteri (quelli speciali, non quelli alfanumerici), e la sbarratura dello zero.

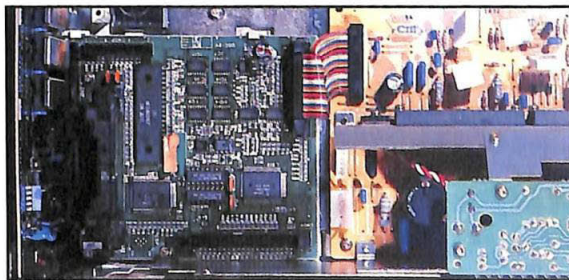
FF	
[LETTER]	LETTER QUALITY
[FONT1]	COUR 10 , 10CPI
[MENU1]	SEE BELOW

LF	
[DRAFT]	DRAFT
[FONT2]	PRSTG12 , 12CPI
[MENU2]	SEE BELOW

NO	Item	MENU1	MENU2
01	QUALITY	LETTER	DRAFT
02	FONT	COUR 10	COUR 10
03	CHR-WIDE	NORMAL	NORMAL
04	CHR-HIGH	NORMAL	2-TIMES
05	CHAR-SPC	12CPI	6 CPI
06	LINE-SPC	6LPI	6LPI
07	EMULATE	JX-80	JX-80
08	ATTRIBT	BOLD	NONE
09	PAGE-LG	11INCH	11INCH
10	COLOR	AUTOSEL	AUTOSEL
11	LEFT-END	1COLM	1COLM
12	TOP-MRGN	1LINE	1LINE
13	LANGUAGE	USA	USA
14	CHAR-SET	SET2	SET2
15	GRAPH-LF		
16	PERFRATN	NO-SKIP	NO-SKIP
17	WIDTH		
18	ZEROFONT	NO-SLASH	NO-SLASH
19	DC3-CODE	ENABLE	ENABLE
20	CR-CODE	CR-ONLY	CR-ONLY
21	LF-CODE	LF&CR	LF&CR
22	RIGHTEND	WRAP	WRAP

NO	Item	HARDWARE
01	PPR-OUT	CNTONLY
02	PRNT-DIR	BI-DIR
03	BUZZER	ON
04	WORD-LG	8-BIT
05	BUFFER	8KBYTE
06	FEEDER	NONE
07	DT-FRMAT	
08	BAUD-RT	
09	PROTOCOL	
10	DSR	

NO	Item	TOF-ADJ
01	ORIGIN	1/6INCH
02	FINE-ADJ	+1/60IN

Il controllo da console, il modo Special

Mentre per impostare i parametri permanenti è necessario riaccendere la stampante e inserire un foglio, per l'abilitazione delle opzioni temporanee è sufficiente entrare nel modo Special premendo il tasto Mode sulla console, senza toccare il foglio già inserito. Dopo aver opportunamente impostato i parametri permanenti relativi alle voci Panel, Menu1 e Menu2, il modo Special è lo strumento di cui ci si serve per operare rapide variazioni (per esempio si può stampare rapidamente una lettera in modo Draft, leggerla su carta per cercare eventuali errori e poi passare a una stampa definitiva in qualità Letter).

In modo Special, la DL3300 comunica con l'utente attraverso i tasti della console e una serie di led: con il tasto Mode si selezionano i significati degli altri due tasti LF e FF, mantenendo il controllo attraverso le scritte e i led che si trovano accanto ai tasti. Agendo quindi sui tasti LF e FF si attivano le opzioni Letter o Draft, Font1 o Font2, Menu1 o Menu2. Ognuna di queste opzioni agisce secondo quanto è stato impostato nei parametri permanenti. Per esempio, si può aver configurato il sistema in modo che attivando l'opzione Draft venga automaticamente selezionata la fonte Hi-Draft, oppure che attivando Font1 la fonte diventi Prestige Elite con spaziatura 6 cpi, e che attivando Menu1 la stampante si configuri secondo i 20 parametri impostati attraverso il modo Setup.

Fra le opzioni di cui è possibile avvalersi da console, la più interessante è proprio la possibilità di mantenere permanentemente in memoria due complete configurazioni comprendenti ciascuna 20 parametri, e attivare una o l'altra con la semplice pressione di pochi tasti.

Sono operazioni sempre molto semplici, che non richiedono nessuna preparazione tecnica, anche se occorre un po' di pazienza per impostare tutti i parametri in modo da ottenere i risultati desiderati.

La prova e l'analisi interna

Come per la DX2300, abbiamo effettuato la prova collegando la DL3300 a un Amiga e selezionando il driver di stampa Epson JX-80.

I risultati, ancora una volta, non sono stati inferiori alle nostre aspettative: sia la meccanica di trascinamento sia la meccanica di stampa hanno ampiamente dimostrato la robustezza e la qualità della DL3300. Come per la DX2300, il sistema di collocazione automatica dei fogli in posizione

di stampa, e il sistema che consente di mantenere in presa il modulo continuo mentre si attiva la frizione, si sono rivelati strumenti precisi ma soprattutto utili.

La qualità di stampa è pari a quella offerta dalla media delle stampanti a 24 aghi, e le fonti disponibili non lasciano che l'imbarazzo della scelta. Il sistema Setup per i parametri permanenti e il sistema Special per i parametri temporanei si sono dimostrati molto efficienti. Interessantissima in particolare l'opportunità di cambiare completamente il comportamento della stampante (fonte, interlinea, spaziatura, stile, qualità...) con la semplice pressione di due tasti.

Come la DX2300, anche la DL3300 consente di montare un kit colore acquistabile separatamente. Non ci siamo soffermati su questo punto perché valgono tutte le considerazioni fatte per la DX2300. L'unica differenza è ovviamente la maggiore densità di punti che la DL3300 riesce a offrire rispetto al modello inferiore. Il prezzo del kit colore non varia molto per le due macchine, e quindi possiamo nuovamente affermare (a maggior ragione) che con un modesto supplemento si può disporre anche di una buona stampante a colori.

Per concludere la prova occorre valutare anche la struttura hardware interna, prestando particolare attenzione ad alcuni aspetti che secondo la nostra esperienza sono fra i più significativi.

Per aprire il cabinet della DL3300 è sufficiente svitare due viti e sollevarne la metà superiore. Come per la DX2300, l'assemblaggio delle parti meccaniche ed elettroniche è stato svolto con estrema cura. Rispetto alla DX2300 notiamo una minor quantità di materiale fonoassorbente (ma il livello di rumorosità è contenuto esattamente come per la DX2300). Su un lato del cabinet si nota un cerchio di fori d'aerazione che dovrebbero sottintendere la presenza di una ventola... che però non esiste. In fase di progettazione, lo si nota anche esaminando la struttura interna, era stato previsto anche l'inserimento di una ventola, ma evidentemente la ventilazione forzata è in seguito stata ritenuta superflua.

La scheda di controllo della stampante è molto sofisticata. Diversamente dalla DX2300, la maggior parte delle componenti elettroniche sono montate direttamente sulle piste del circuito stampato, risparmiando così notevole spazio. Alloggia un connettore con inseritore in plastica per le cartucce d'espansione, e il connettore per la scheda dell'interfaccia, sulla quale, a differenza della DX2300, è presente anche il firmware di gestione della stampante. L'interfaccia è facilmente sostituibile per adattare la stampante alla connessione seriale o parallela del computer. La cura hardware è testimoniata anche da piccoli particolari come il panno spugnoso che non permette l'ingresso d'impurità attraverso la feritoia per l'inserimento delle cartucce d'espansione che si trova sul cabinet.

Siamo quindi di fronte, come per la DX2300, a una macchina curata in ogni dettaglio. Come per il modello inferiore tutti i connettori sono facilmente scollegabili, e le parti meccaniche sottoposte a più intense sollecitazioni sono montate su supporti di gomma per ridurre le vibrazioni impresse al cabinet soprattutto in alta velocità. Con ogni probabilità l'elevata qualità dei materiali dipende anche dalla produzione su larga scala.

Conclusioni

La DL3300 è una stampante di buona qualità. Per quanto sia il modello inferiore della famiglia Fujitsu a 24 aghi, la sua solidità, la cura di ogni dettaglio, la praticità d'uso e le soluzioni tecniche adottate – come il modo Special di stampa, la varietà di fonti, stili e spaziature, la possibilità di non togliere mai il modulo continuo – rendono questa stampante particolarmente accattivante sotto il profilo della semplicità e versatilità d'impiego. La DL3300 offre una facilità d'uso non comune, che la rende uno strumento ideale in ambito professionale, e che farà la gioia di chi è alla ricerca di un buon prodotto da affiancare al proprio computer ed è disposto a pagare il prezzo (in questo caso giustificato) della qualità. ■

DATI TECNICI

	DL3300	DL3400
Testina di stampa	24 aghi a impatto	
Velocità di stampa	288 cps Hi-Draft a 12 cpi 240 cps Hi-Draft a 10 cpi 216 cps Draft a 12 cpi 180 cps Draft a 10 cpi 72 cps Letter a 12 cpi 60 cps Letter a 10 cpi	
Densità di stampa	2,5/3/5/6/10/12/15/17,1/18/20 cpi e proporzionale	
Caratteri per linea	80 cpl 96 cpl 120 cpl 136 cpl 144 cpl 160 cpl	136 cpl 163 cpl 204 cpl 232 cpl 244 cpl 272 cpl
Fonti carattere	3 residenti, 8 con cartucce d'espansione, 2 caricabili in RAM	
Buffer di stampa	1/4, 2, 8, 24K	
Trascinamento	A trattore per moduli perforati A frizione per fogli singoli	
Emulazioni	IBM Graphics Printer IBM Proprinter Epson FX-80/JX-80 Diablo 630 API Fujitsu DPL24C/D/I	
Interfaccia di serie	Parallela Centronics o seriale	
Dimensioni	120 x 438 x 345 mm	120 x 580 x 345 mm
Peso	9 kg	12 kg
Prezzo al pubblico	L. 1.732.000 + IVA (p) L. 1.806.000 + IVA (s) (p - parallela, s - seriale)	L. 1.995.000 + IVA (p) L. 2.079.000 + IVA (s)

Optional

	DL3300	DL3400
Caricatore automatico di fogli singoli	L. 500.000 + IVA	L. 550.000 + IVA
Secondo cassetto (costo aggiuntivo)	L. 400.000 + IVA	L. 400.000 + IVA
Kit colore	L. 200.000 + IVA	L. 200.000 + IVA
Cartucce fonti carattere		L. 130.000 + IVA

Importatore

Fujitsu Italia Spa
Via Melchiorre Gioia, 8 - 20124 Milano (tel. 02/6572741/2/3/4/5)

UN FOGLIO ELETTRONICO PER IL MONDO DI GEOS

GeoCalc fa entrare nel mondo di GEOS l'analisi numerica: un foglio elettronico facile da usare, dotato di caratteristiche interessanti, come gli indirizzi simbolici e i raggruppamenti di caselle, ma privo di alcune importanti funzioni

di Luca Giachino

Il foglio elettronico, o tabellone, è una delle pietre miliari del mondo informatico. Non c'è computer, dai mainframe ai mini, per il quale non ne sia stata realizzata almeno una versione. Le sue origini risalgono al 1978, anno in cui nasceva *VisiCalc*, il primo tabellone computerizzato. Fu un successo strepitoso in tutto il mondo, una testimonianza dell'estrema validità e praticità d'uso del foglio elettronico nelle analisi numeriche.

Il suo fascino è implicito nell'idea di partenza. Si pensi a quanto numerose sono le applicazioni di uno strumento quasi banale come un tabellone suddiviso in righe e colonne: può registrare le varie fasi di un'analisi, una programmazione di lavoro, un bilancio di previsione, la situazione del campionato di calcio... In questa forma, però, il tabellone è uno strumento che permette al massimo di crearsi un'idea complessiva di una situazione articolata, oppure funge da promemoria, ma è del tutto inerte.

Se però i dati numerici presenti nelle varie caselle potessero dipendere, secondo un'opportuna relazione matematica, dai dati contenuti nelle altre caselle, il tabellone potrebbe ricalcolare in modo autonomo i valori di tutte le caselle ogni volta che vengono cambiati alcuni dati "chiave". Sarebbe in un certo senso un tabellone intelligente.

Su questa considerazione nacque il primo "tabellone computerizzato" o foglio elettronico, un'idea che, grazie alla sua semplicità, ancor oggi resiste quasi immutata: l'utente vede sullo schermo del computer un insieme di caselle e in queste caselle può inserire testi, valori numerici e formule matematiche. Le caselle sono individuate univocamente da indirizzi riga/colonna, e quando si varia un dato il foglio elettronico provvede a ricalcolare tutte le caselle che sono in relazione diretta o indiretta con la casella alterata.

Uno degli aspetti che ne hanno garantito il successo è l'immediata associazione schermo/foglio di quaderno che riesce spontanea anche all'utente più diffidente nei confronti del computer (tra l'altro da quest'immagine trae ispirazione il nome anglosassone del foglio elettronico: spreadsheet, foglio aperto). Sul "foglio" si possono distribuire tutti i parametri di un problema, delegando i calcoli all'instancabile computer. Oppure, come accade nella maggior parte dei casi, lo si può utilizzare per i bilanci, per i calcoli millesimali delle quote dovute dai vari condomini, per il ricalcolo dei prezzi in seguito a un aumento generale delle spese...

Insomma, il foglio elettronico è diventato un elemento indispensabile per una biblioteca software, al pari del word processor e del database.

Un mondo completo

Nel mondo di *GEOS*, che ha l'ambizione di costituire un ambiente di lavoro completo per gli utenti del C-64 e del C-128, non poteva certo mancare un foglio elettronico: e infatti tra i prodotti marchiati Berkeley è presente anche *geoCalc*.

Una piccola premessa: quando si parla di *GEOS* si parla di C-64, quindi nessuno si aspetta miracoli di memoria e di velocità. Nel caso dello spreadsheet *geoCalc* diciamo subito che vi sono alcuni difetti da segnalare, ma anche alcuni pregi che ci permettono di definirlo un prodotto più che dignitoso. Con ogni probabilità gli utenti di *GEOS* lo accoglieranno con favore anche se non con entusiasmo.

Grazie all'interfaccia utente

user-friendly, tipica di *GEOS*, e la sua veste grafica avanzata (menu, icone, mouse...), *geoCalc* riesce a far convivere un'ottima praticità d'impiego con un aspetto gradevolissimo, anche se – come vedremo in seguito – è privo di alcune importanti caratteristiche che si ritrovano quasi sempre in pacchetti analoghi.

Per alcune utilizzazioni limitate, *geoCalc* può offrire reali soddisfazioni. È ad esempio perfetto per chi desidera realizzare un piccolo bilancio familiare, valutare il rapporto spese/guadagni di una piccola azienda, risolvere un'equazione con metodi iterativi...

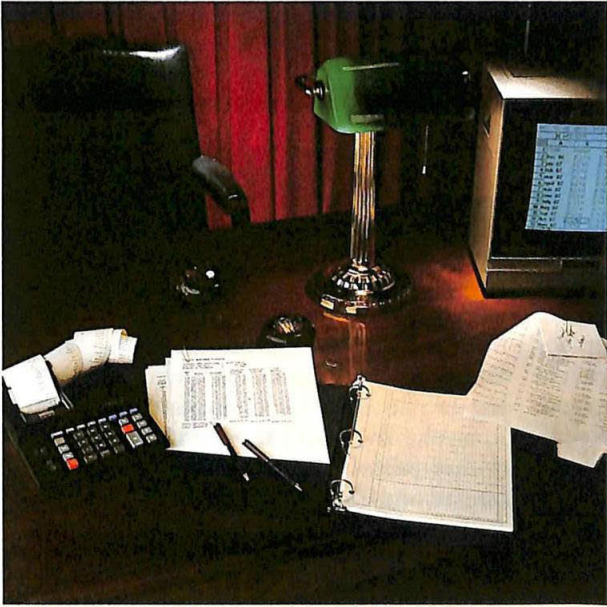
GeoCalc

Il foglio gestito da *geoCalc* è una matrice bidimensionale suddivisa in 28.672 caselle (256 righe per 112 colonne). Anche se un numero così elevato di caselle nella maggior parte dei casi è eccessivo, va notato che prodotti analoghi (parliamo sempre di prodotti per il C-64) non vanno oltre le 16 mila caselle, e in qualche caso questa maggiore estensione del foglio si rivela l'asso nella manica di *geoCalc*. Quando il programma viene mandato in esecuzione, sullo schermo appaiono cinque colonne e tredici righe, per un totale di 65 caselle. Ma anche se queste 65 caselle occupano quasi l'intero schermo, costituiscono solo lo 0,26 per cento dello spazio disponibile.

GeoCalc permette di cambiare la larghezza delle caselle a piacere, anche solo per un particolare gruppo di colonne. Si può visualizzare da un minimo di 13 a un massimo di 156 caselle (dallo 0,045 per cento allo 0,54 per cento del foglio). Da queste semplici considerazioni è già possibile farsi un'idea della vastità di spazio a disposizione.

L'interfaccia utente è come al solito molto efficiente. Grazie ai menu, alle icone e ai box di dialogo, permette di non abbandonare mai il foglio durante l'attivazione delle opzioni disponibili: per immettere un dato o una formula in una casella è sufficiente selezionarla con il mouse e scrivere il dato nell'apposita "area inserimento" che si trova sotto il menu principale.

Un esempio pratico

Per capire meglio come si usa il programma, e anche a che cosa può servire in pratica un foglio elettronico, conviene ricorrere a un caso pratico. Un bilancio familiare, per esempio: se si hanno sottomano tutte le bollette e i conti, con un po' di pratica si riesce a trasferirlo sul foglio elettronico in pochi minuti, in modo da poter poi operare tutte le modifiche, le elaborazioni e i confronti che appariranno necessari.

Iniziamo intestando una serie di righe con i diversi tipi di spese mensili (telefono, affitto, benzina, divertimenti...) e 12 colonne che rappresentano i mesi dell'anno. Possiamo evidenziare maggiormente questi testi attivando l'opzione Bold, operazione che può essere effettuata per una sola casella o per un insieme di caselle. Tramite il mouse si può facilmente selezionare una riga, una colonna, oppure una regione del foglio, in modo che ogni opzione attivata, come Paste o Bold, si estenda automaticamente a tutte le caselle selezionate.

A questo punto abbiamo creato una matrice di caselle dotata d'intestazione orizzontale (i mesi

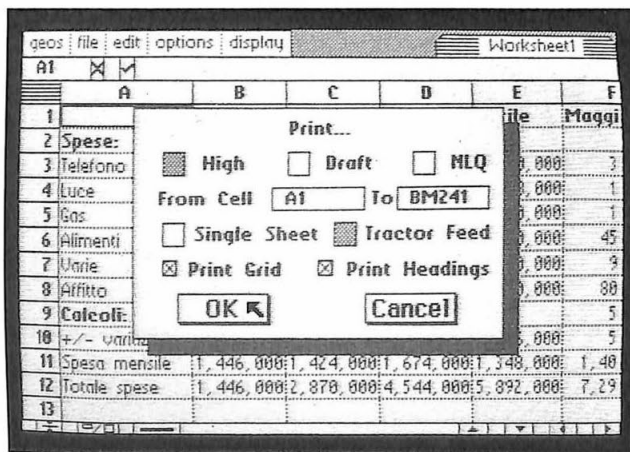
dell'anno) e intestazione verticale (le principali spese mensili). Per ogni mese si possono indicare i costi sostenuti per ogni voce, ma se si fermasse qui, *geoCalc* sarebbe ancor meno utile di un comunissimo foglio di carta.

A questo punto, dopo aver aggiunto un testo che potrebbe essere "Spesa mensile" in corrispondenza della riga prevista per i risultati, non rimane che dotare *geoCalc* di una certa capacità di elaborazione. Per farlo, però, dobbiamo immettere alcune formule: inseriamo nella casella corrispondente a gennaio la formula che somma tutti i costi mensili, cioè il contenuto di tutte le caselle relative alle spese fatte in gennaio. In questo modo, quasi per magia, nella casella intersezione della riga dei totali con la colonna di gennaio appare la spesa totale del mese. Per inserire la formula, *geoCalc* offre svariate possibilità. È possibile scriverla direttamente, citando l'indirizzo riga/colonna delle caselle da sommare, oppure, trattandosi di una semplice somma, selezionare con il mouse le caselle da sommare: *geoCalc* completerà automaticamente la formula matematica corrispondente separando l'indirizzo di ogni casella dall'operatore di addizione "+".

La somma dei costi mensili la desideriamo per tutti i mesi dell'anno, ma riscrivere la stessa formula nelle altre caselle della riga dei totali può essere fonte di errori. Del resto, da un punto di vista topologico (cioè identificando le caselle secondo le loro posizioni rispetto alla casella attuale) la formula è sempre uguale, visto che si può riassumere nel comando: «Somma i contenuti di quattro caselle consecutive sopra a quella dove ti trovi adesso, partendo dalla terza». Gli indirizzi delle singole caselle variano di mese in mese (gli indirizzi di riga sono immutati per tutti i mesi, mentre gli indirizzi di colonna variano di mese in mese) e quindi la pura copia di questa formula per tutti i mesi darebbe come risultato sempre la somma dei costi rilevati in gennaio. Fortunatamente però i fogli elettronici sono in grado di gestire sia indirizzi assoluti sia indirizzi relativi, e *geoCalc* non è da meno. Quindi nel nostro caso è opportuno che la prima formula sia espressa con riferimenti relativi, in modo che copiandola nelle opportune caselle degli altri mesi dell'anno, gli indirizzi dei suoi addendi mantengano la corrispondenza topologica. Se dopo aver copiato la formula in tutte le caselle dei dodici mesi si seleziona una di queste caselle, si nota che la formula (che appare nell'area inserimento, sotto il menu principale) è analoga alla precedente, ma gli addendi sono quelli della colonna considerata.

I riferimenti assoluti servono invece quando in una formula si vogliono individuare sempre le stesse caselle, a prescindere dalla posizione in cui la formula viene collocata nel foglio.

Per esempio, se nel precedente bilancio s'individua un parametro che rimane invariato per tutti i mesi dell'anno, risulta conveniente associarlo a una casella a parte e indicarlo nelle formule di calcolo con un indirizzo assoluto. Ogni volta che questo dato viene alterato, si ottiene l'automatica variazione di tutti i risultati prodotti dalle funzioni coinvolte.

La formula della spesa mensile è stata copiata per tutti i mesi

Il box delle opzioni di stampa offerte dallo spreadsheet geoCalc

GeoCalc diventa intelligente

Il semplice esempio che abbiamo proposto finora può diventare molto più complesso: possiamo

aumentare notevolmente il numero di dati che *geoCalc* deve gestire e di conseguenza e i risultati che deve calcolare. Possiamo per esempio avvalerci della vastità del foglio per organizzare tanti piccoli bilanci parziali, differenziandoli magari per categorie di costi. A questi si possono aggiungere aree del foglio che calcolino dati statistici per individuare discontinuità rilevanti nella curva delle spese, fino a rendere il bilancio molto elaborato. Spostando la finestra di *geoCalc* sull'area desiderata si possono visualizzare in ogni momento i risultati o anche immettere nuovi dati. *GeoCalc* gestisce tutti i legami tra le varie parti del foglio in modo che i risultati siano sempre aggiornati.

Se poi diventasse interessante visualizzare contemporaneamente alcuni dati appartenenti ad aree diverse, è sufficiente ricorrere all'icona che divide la finestra di *geoCalc* in due parti. Infine, quando i dati sono stati correttamente inseriti, si passa alla stampa parziale o totale del foglio. Fortunatamente *geoCalc* prevede anche la stampa draft, meno raffinata di quella in alta risoluzione, ma molto più rapida. Infatti, per un'applicazione di questo genere, la stampa può essere fondamentale, e se la quantità dei dati è elevata la rapidità diventa un fattore di primaria importanza.

A parte interessanti caratteristiche come il raggruppamento di caselle e le funzioni di cut & paste, in questo esempio abbiamo analizzato le caratteristiche di *geoCalc* comuni alla stragrande maggioranza dei fogli elettronici. Vediamo ora più da vicino le peculiarità di *geoCalc*.

La finestra mostra due aree diverse del foglio elettronico

La selezione dei gruppi di caselle

Quando l'utente attiva un'opzione tramite i menu, come il formato di output per i dati numerici, lo stile, l'allineamento, la copia e così via, *geoCalc* agisce sempre sul gruppo di caselle selezionato in quel momento. Un "gruppo" può essere composto tanto da un'unica casella quanto dall'intero foglio. Quando si seleziona una casella sullo schermo, si crea in pratica il più semplice raggruppamento consentito da *geoCalc*. Le opzioni che vengono attivate agiscono solo su quella casella. Se si desidera estendere l'opzione a più caselle, si deve solo selezionare con il mouse la relativa regione del foglio, un'operazione molto semplice. La regione può contenere caselle disposte su una sola riga o su una sola colonna, ma anche insiemi di colonne o insiemi di righe. In quest'ultimo caso, le caselle selezionate appaiono sullo schermo come un rettangolo in negativo.

Talvolta può essere utile selezionare tutte le caselle di una colonna o di una riga del foglio. *GeoCalc* permette, premendo il pulsante del mouse sull'intestazione della colonna o della riga, la selezione di tutte le sue caselle. Infine, è possibile selezionare l'intero foglio attivando un'apposita icona. A questo proposito è bene ricordare che se non si dispone di un RAM disk, l'estensione di una semplice opzione all'intero foglio costituisce per *geoCalc* un lavoro molto lungo, dal momento che in memoria vengono mantenuti i dati relativi a matrici di 16 x 16 caselle, e quindi si rendono necessari continui accessi al disco per raggiungere tutte le caselle del foglio. In generale, tutte le opzioni che coinvolgono gruppi di caselle molto estesi costringono a lunghi tempi d'attesa. Per esempio, il tempo d'attesa diventa interminabile se l'opzione selezionata è il Paste di una formula da riprodurre in un elevato numero di caselle. Infatti, *geoCalc* compie un accesso al disco per ogni casella presente nel gruppo (ovvero legge il contenuto del file Calc Scrap per immetterlo nella casella). Si tratta di un procedimento molto poco efficiente, perché non sarebbe stato difficile fare in modo che il programma mantenesse in memoria il file Calc Scrap, senza doverlo caricare da disco per ogni casella.

I raggruppamenti di caselle servono anche per guidare l'input dell'utente. Se per esempio intendiamo immettere i dati in una matrice di 4 x 4 caselle, possiamo selezionarle come un gruppo unico, in modo che *geoCalc* si sposti automaticamente alla casella successiva della matrice al termine di ogni inserimento.

I riferimenti assoluti e relativi

GeoCalc permette d'identificare una casella tramite un riferimento assoluto nel foglio, o relativo a un'altra casella. Il riferimento assoluto individua sempre la medesima casella, mentre quello

relativo stabilisce una corrispondenza topologica fra la casella in cui ci si trova e quella cercata. Una formula può contenere qualsiasi insieme di riferimenti assoluti e relativi, secondo le specifiche esigenze.

La differenza fra i due tipi di riferimenti assume particolare importanza quando una formula viene copiata su un insieme di caselle. Infatti i riferimenti assoluti rimangono invariati, mentre i riferimenti relativi vengono alterati da *geoCalc* a ogni copia, in modo che vengano mantenute le relazioni topologiche.

Un riferimento può anche essere "parzialmente assoluto" (o "parzialmente relativo"). Se per esempio si desidera che ogni elemento di una colonna sia uguale a quello contenuto nella prima casella della stessa colonna (che chiamiamo A1), occorre procedere nel modo seguente: nella casella A2 si scrive la formula:

= A$1

dove A = colonna relativa, $1 = riga assoluta (il carattere $ stabilisce che l'indirizzo di colonna o di riga che segue è da ritenere assoluto). Con questa formula, la casella assume il contenuto della casella A1. Se ora si copia questa formula nelle caselle A3, A4 e A5, ognuna di queste assume a sua volta il contenuto della prima casella. Se la stessa formula viene copiata nelle caselle delle colonna B, *geoCalc* cambia automaticamente l'indirizzo relativo A in B in modo che venga individuata sempre la prima casella della colonna in cui ci si trova (viene cioè mantenuta la relazione topologica), ma lascia inalterato l'indirizzo di riga $1 in modo che la formula individui sempre la prima riga del foglio.

I nomi delle caselle

GeoCalc consente di attribuire a qualsiasi riferimento un nome. Attivando l'opzione Define Name del menu Options, appare un box di dialogo nel quale *geoCalc* visualizza i nomi definiti fino a quel momento. Appaiono inoltre due finestre: la prima contenente l'indirizzo della casella in cui si

Il box che permette di assegnare un nome a una casella

trova il cursore e la seconda vuota. Se ci si limita a scrivere un nome nella casella vuota, questo verrà assegnato (come assoluto) alla casella corrente; se invece, oltre che digitare un nome nella finestra vuota (per esempio "Sotto"), si compie l'ulteriore operazione di cancellare l'indirizzo contenuto nell'altra finestra (per esempio B11) e lo si sostituisce con un altro (per esempio B12), il nome "Sotto" verrà attribuito da questo momento in poi alla casella immediatamente inferiore a quella in cui ci si trova (si mantiene cioè la stessa posizione relativa tra B11 e B12). In questo modo, ogni volta che in una formula si vuole indicare la casella inferiore a quella attuale, è sufficiente scrivere Sotto, senza dover ricorrere all'indirizzo assoluto di quella casella. L'ultima possibilità offerta da *geoCalc* in questo senso è la completa disattivazione dei nomi: il programma visualizza i riferimenti solo attraverso i loro reali indirizzi.

I nomi sono particolarmente utili nella formulazione di un problema e nelle fasi di configurazione del foglio: anziché ricordare a memoria che la percentuale di sconto si trova nella casella H100, l'unico sforzo diventa quello di ricordare il nome Sconto_Attuale: la gestione simbolica del foglio elimina molte difficoltà ed è una delle qualità più interessanti di *geoCalc* che non sia condivisa da tutti gli spreadsheet.

La praticità

GeoCalc mette a disposizione dell'utente diverse icone che permettono di svolgere in modo automatico le operazioni più frequenti. Nella parte inferiore dello schermo ci sono sei icone: una per dividere la finestra principale in due parti indipendenti, in grado di visualizzare due diverse aree del foglio. Un'altra per ridurre l'area inserimento alle sue normali dimensioni, quando la lunghezza del testo o della formula ne ha esteso la visualizzazione anche sopra la finestra di schermo (il limite per il contenuto di una singola casella è di circa 200 caratteri, mentre l'area inserimento ne visualizza in condizioni normali soltanto una trentina). Le ultime quattro, infine, servono per muovere la finestra di schermo nelle quattro direzioni sul foglio. Fra queste icone, forse un po' troppo piccole per essere davvero comode, si trova anche un piccolo box che *geoCalc* visualizza in negativo quando è impegnato in qualche operazione che non gli consente di ricevere input

dall'utente. Secondo il manuale, questa condizione si verifica soltanto durante gli accessi al disco e durante il ricalcolo delle caselle. Ma in realtà la stessa cosa avviene anche in altre occasioni, per esempio se l'utente richiede l'attivazione del Bold su un grande numero di caselle: tutto quello che si può fare è spostare il mouse, ma se si cerca di attivare una qualunque parte dello schermo non accade nulla. Purtroppo in casi come questo il piccolo box non segnala nulla, lasciando l'utente in un profondo stato di angoscia al pensiero che qualcosa sia andato storto e che il programma abbia deciso di chiudere bottega. Sarebbe stato meglio un flag che in qualche modo visualizzasse sullo schermo gli eventuali periodi d'inattività di MainLoop, ovvero le situazioni in cui il sistema non può prestare attenzione all'utente (per ottenere maggiori dettagli sulla struttura interna del sistema operativo *GEOS*, si veda la serie di articoli "Anatomia di un sistema operativo" apparsi su *Commodore Gazette* a partire dal numero 1/88, e la *Guida ufficiale alla programmazione di GEOS* edita dalla IHT Gruppo Editoriale).

Lo scroll del foglio nella finestra è decisamente molto lento. Questa lentezza è da attribuire in parte all'alta risoluzione impiegata da *GEOS* per lo schermo e in parte alla necessità di accedere al disco per caricare i dati relativi a zone del foglio non presenti in memoria. Si tratta di una deficienza che diventa pesante quando uno schermo non è più sufficiente per contenere tutte le informazioni. Un'espansione RAM migliora comunque la situazione; la 1764 (256K) rappresenta, come abbiamo già avuto modo di dire, il miglior compromesso prezzo/prestazioni che si possa ottenere in ambiente *GEOS*.

Lo scroll può anche avvenire muovendo il mouse su uno dei bordi dello schermo, o attivando uno sprite di spostamento che permette di muoversi lungo il foglio con discreta rapidità.

Sopra la finestra di schermo, sono presenti due icone per l'inserimento dei dati e la cosiddetta area inserimento, dove appaiono il cursore e il testo contenuto nella casella selezionata. Agendo in quest'area l'utente immette o modifica i dati presenti nella casella. Se s'inserisce un testo abbastanza lungo, *geoCalc* sconfina nelle caselle adiacenti, a meno che non contengano già qualcosa. Questa caratteristica permette d'inserire nel foglio elettronico testi esplicativi di qualunque lunghezza.

GeoCalc consente di variare il layout dei dati nelle caselle. Si possono abilitare le opzioni Bold (neretto) e Italic (corsivo), e per i numeri è possibile stabilire il numero di decimali da visualizzare (si ricordi che nei Paesi anglosassoni per separare il numero intero dai suoi decimali si usa il punto, mentre la virgola separa le migliaia). Inoltre si può scegliere l'allineamento (sinistro, centrato, destro) dei testi e dei numeri all'interno delle singole caselle. Come tutte le opzioni, anche queste vengono sempre considerate valide per l'intero gruppo di caselle selezionato.

Il menu principale mette a disposizione, oltre alle opzioni relative allo spreadsheet, anche alcune funzioni generiche che sono presenti in tutte le applicazioni della Berkeley (inspiegabilmente manca la funzione Recover, sempre utile per rimediare ai disastri). Queste funzioni permettono per esempio di attivare i vari modi di visualizzazione, le funzioni di cut & paste e così via.

L'input dei dati è abbastanza articolato, e per sfruttarlo appieno occorre imparare a memoria alcune combinazioni di tasti grazie alle quali si può cambiare facilmente la casella selezionata senza adoperare il mouse (per esempio per chiudere un inserimento e iniziarne uno nuovo).

GeoCalc e la matematica

Tocchiamo ora uno degli aspetti più importanti dei fogli elettronici: la capacità di calcolo e le funzioni predefinite. *GeoCalc* è in grado di effettuare le operazioni algebriche arrivando fino a 12 cifre significative, mentre per le operazioni trigonometriche, i radicali, i logaritmi e così via *geoCalc* arriva a 9 cifre significative. I valori numerici possono variare fra 10^{-65} e 10^{62} nel primo caso, 10^{-37} e 10^{38} nel secondo caso. Il numero di cifre significative dovrebbe essere più che sufficiente nella maggior parte dei casi.

Oltre ai vari operatori algebrici, *geoCalc* mette a disposizione una libreria di funzioni matematiche, statistiche, finanziarie e scientifiche. Le funzioni matematiche comprendono il valore assoluto, il valore intero, il troncamento della parte decimale, la generazione di un numero casuale compreso fra 0 e 1, la radice quadrata. Le funzioni statistiche calcolano la media aritmetica di una serie di caselle, identificano il valore massimo e il valore minimo in un insieme, la somma di una serie di caselle. Le funzioni finanziarie permettono di calcolare, in regime d'interesse composto, il capitale da depositare per ottenere un certo montante, in un dato tempo e con un dato tasso d'interesse. Permette inoltre, utilizzando le formule inverse (anch'esse predefinite) di scegliere come incognita montante, tempo o tasso d'interesse, e calcolarle in funzione delle altre tre grandezze. Si tratta comunque di una sola formula, benché moltiplicata apparentemente per quattro, e – come ben sa anche uno studente di ragioneria – non ci si ferma certo a questo punto, con la matematica finanziaria.

Le funzioni scientifiche consentono il calcolo del logaritmo neperiano e decimale, dell'esponenziale, del π, delle funzioni trigonometriche principali (il manuale riporta le formule per ricavare anche quelle iperboliche). Infine, *geoCalc* permette di associare una funzione di valore nullo a una casella che dovrà comparire

continua a pagina 81

FOGLI ELETTRONICI: ALCUNE APPLICAZIONI INCONSUETE

Formulare matematicamente un problema in modo che un computer possa risolverlo non è sempre un'impresa semplice. I computer adottano linguaggi estremamente primitivi rispetto ai nostri sistemi di comunicazione e mancano d'intuito, quindi allo sforzo di formulare il problema con precisione si aggiunge inevitabilmente quello di trasformarlo in un programma eseguibile.

È ben noto il fastidio che comporta occuparsi dei dettagli più irrilevanti, quando l'unica cosa davvero importante dovrebbe essere individuare la soluzione. Del resto, non sempre è possibile avvalersi di un traduttore (o programmatore) che interceda per noi presso il computer.

Quando il problema rientra in una categoria ben precisa, è possibile procurarsi un programma specifico, evitando il fastidio di doversi trasformare in programmatori. Ma quando i problemi che di volta in volta si devono risolvere sono molto diversi fra loro e non consentono d'individuare sul mercato un unico pacchetto applicativo in grado di affrontarli, l'utente ha di fronte – o almeno così sembrerebbe – soltanto due strade: studiarsi l'arte della programmazione o pagare a caro prezzo programmi che gli serviranno soltanto una o due volte.

Il foglio elettronico

Grazie a questo particolare tipo d'applicazione, tutti i problemi si superano con un compromesso fra la massima elasticità offerta dalla programmazione e la massima rigidità offerta dai pacchetti specializzati.

Un foglio elettronico, infatti, richiede uno sforzo minimo per essere configurato e permette di risolvere, sperimentare e verificare una grande varietà di problemi finanziari, scientifici, statistici, decisionali e di simulazione: ogni situazione in cui il problema può essere formulato in termini matematici.

Il principio di base è la possibilità di assegnare a ogni casella una formula matematica. Il dato visualizzato nella casella rispecchia sempre il risultato prodotto dall'elaborazione della formula. In genere le formule dipendono anche dai dati presenti nelle altre caselle del foglio. Configurare un foglio elettronico significa creare tutte le relazioni matematiche che descrivono il problema: per ogni nuovo problema o ulteriore sviluppo non si fa altro che intervenire sulla configurazione del foglio, o crearne una nuova.

Come funziona

Dal punto di vista dell'utente, il foglio elettronico opportunamente configurato è un insieme di relazioni matematiche che vengono risolte tutte nello stesso istante; non esiste un ordine di priorità. Generalmente i fogli calcolano prima tutte le formule, poi confrontano i risultati con quelli che ottengono in una seconda fase di calcolo. Se anche in una sola casella i due risultati sono diversi, il programma provvede a rielaborare l'intero foglio. Questo procedimento di ricalcolo continua fino a quando il valore di ogni casella è uguale (entro l'approssimazione prevista) al nuovo valore ottenuto con il successivo ricalcolo del foglio. Per quanto ridotto all'osso, questo è il procedimento tramite il quale un foglio si stabilizza, cioè smette di effettuare calcoli. Ogni foglio elettronico prevede poi alcuni algoritmi per ottimizzare il procedimento di ricalcolo prendendo in considerazione ogni volta soltanto le caselle effettivamente implicate.

Questa struttura è di per sé affascinante, perché – in un certo senso – affronta i problemi "in parallelo". Nella maggior parte dei linguaggi di programmazione, la soluzione di un problema viene raggiunta attraverso un algoritmo, cioè una serie di passi ben precisi da compiere in sequenza. Un algoritmo potrebbe essere l'indicazione che si dà a un passante per raggiungere una piazza: «Prosegua fino all'incrocio, giri a destra e poi dritto fino alla farmacia...». Una struttura "in parallelo", invece, cercherebbe di descrivere la posizione del negozio fornendo una serie d'indicazioni che non devono necessariamente essere valutate in sequenza.

Ovviamente non tutti i problemi possono essere risolti sostituendo a un algoritmo una struttura parallela, e non sempre è detto che una soluzione parallela sia migliore di una sequenziale.

In genere i fogli elettronici sono concepiti per rappresentare problemi finanziari. Si prestano molto bene a questo genere di applicazioni in quanto generalmente i dati che costituiscono un problema finanziario sono molti e non sempre si riesce a identificare le variabili indipendenti; si tende quindi a effettuare stime diverse, cambiando di volta in volta la variabile indipendente. Anche se questo è uno dei settori in cui il foglio elettronico viene maggiormente usato, è affascinante pensare a quanti argomenti scientifici possono essere rappresentati con questa tecnica che, parlando di argomenti matematici, e volendo usare un tono un po' sostenuto potremmo definire "tecnica della matrice bidimensionale elettronica".

Un foglio matematico

Vediamo ora alcuni esempi d'impiego del foglio elettronico in generale (e di *geoCalc* in particolare) nell'inconsueto settore della matematica, anziché in quello economico/amministrativo.

Iniziamo con un esempio relativamente semplice. Se in *geoCalc* selezioniamo la casella A3 e le associamo la formula:

= A1 + A2

otteniamo in A3 la somma delle due caselle soprastanti. Associamo ora il valore numerico 1 a A1 e 2 a A2; in A3 otterremo 3. Ora copiamo la formula che definisce il contenuto di A3 anche in A4, A5, A6, A7 e A8. Avendo utilizzato indirizzi relativi, per ognuna di queste caselle verrà calcolata la somma delle due caselle superiori: senza sviluppare nessun algoritmo, ma semplicemente definendo una serie di operazioni simili su caselle diverse, abbiamo descritto una struttura parallela per calcolare i primi otto valori della serie di Fibonacci. Infatti i numeri che otteniamo nella colonna A sono 1, 2, 3, 5, 8, 13, 21. Ovviamente avremmo potuto ottenere la stessa serie in moltissimi altri modi, anche con un semplicissimo algoritmo in Basic:

```
10 m=1:n=2
20 print m:print n
30 m=m+n
```

```
40 n=n+m
50 goto 20
```

ma l'impiego del foglio, per quanto il problema sia molto semplice, ha il vantaggio di permettere anche a chi non sa nulla di programmazione di arrivare a un risultato.

Con un procedimento simile possiamo facilmente calcolare i coefficienti delle potenze di ordine "n" di un binomio. I coefficienti sono calcolabili con il famoso triangolo di Tartaglia,

(0) 1
(1) 1 1
(2) 1 , 2 1
(3) 1 3 3 1
(4) 1 4 6 4 1
(5) 1 5 10 10 5 1
...

nel quale ogni riga individua i coefficienti da impiegare nello sviluppo della potenza del binomio, il cui esponente è indicato nella prima colonna. Questa serie di coefficienti, che si può estendere a piacere, si può ricavare con *geoCalc* associando a ogni casella la somma tra il contenuto della casella superiore e quello della casella superiore sinistra. In pratica si deve riempire la colonna A di zeri (il numero di righe dipende dalla profondità del triangolo che si vuole ottenere), e inserire il valore 1 nella casella B1. Nella casella B2 si inserisce la formula:

= B1 + A1

Questa formula dev'essere poi copiata in tutte le altre caselle della colonna B, in tutte le caselle della colonna C a partire da C2, nella colonna D a partire dalla casella D3 e così via fino all'ultima colonna e l'ultima riga. In questo modo *geoCalc* produce sullo schermo i coefficienti del triangolo di Tartaglia senza avvalersi di un algoritmo, ma di un'informazione che lega ogni coefficiente con quelli della riga precedente.

Proviamo ora a complicare un po' le cose. Talvolta, magari durante la preparazione dell'esame di Analisi I all'università, ci si imbatte in un'equazione che non è possibile risolvere con i metodi che l'algebra mette a disposizione. Questa condizione può per esempio verificarsi quando si cercano le intersezioni con l'asse x della funzione $y = f(x)$, cioè le soluzioni dell'equazione $f(x) = 0$, e la funzione $f(x)$ è trascendente. Un'equazione di questo tipo è, per esempio:

$$\sqrt{\ln x + 2x} - x = 0$$

che, appunto, non si può risolvere con metodi algebrici. Con *geoCalc*, però, possiamo applicare un metodo di risoluzione iterativo. Il procedimento consiste nello scindere la formula in un sistema di due equazioni: $y = g(x)$ e $y = x$. Dal punto di vista geometrico, si tratta quindi di cercare i punti d'intersezione tra la curva $g(x)$ e la bisettrice del primo quadrante:

$$y = \sqrt{\ln x + 2x}$$
$$x = y$$

geos | file | edit | options | display | iterativo

B5 =SQRT(LN(B6)+B6*2)

	A	B	C
1		Tabellone elettronico per la soluzione	
2		di un'equazione con il metodo iterativo.	
3		Equazione: X = SQR(LN(X) + 2*X)	
4			
5	X1 nuovo	2.18645343	X1 = f(X2)
6	X2 attuale	2.03503323	X2 = X1
7	Delta	0.1514202	Delta = |X2-X1|
8			
9		L'equazione e' di tipo trascendente,	
10		e non e' risolvibile con nessun	
11		metodo algebrico.	
12		Soluzione	2.03503323
13			

Sopra: una fase intermedia. Sotto: la soluzione del problema

geos | file | edit | options | display | iterativo

A1

	A	B	C
1		Tabellone elettronico per la soluzione	
2		di un'equazione con il metodo iterativo.	
3		Equazione: X = SQR(LN(X) + 2*X)	
4			
5	X1 nuovo	2.36393887	X1 = f(X2)
6	X2 attuale	2.36393887	X2 = X1
7	Delta	0	Delta = |X2-X1|
8			
9		L'equazione e' di tipo trascendente,	
10		e non e' risolvibile con nessun	
11		metodo algebrico.	
12		Soluzione	2.36393887
13			

Per risolverlo, applichiamo il seguente procedimento iterativo: si stima un valore di x plausibile (x dev'essere perlomeno maggiore di zero, in modo che il logaritmo sia calcolabile), e si calcola il valore della y nella prima funzione $y = g(x)$. Questo valore diventa il nuovo valore di x, grazie alla seconda equazione, e viene ancora una volta inserito nella prima, e così via fino a quando il valore di x, a ogni nuova iterazione, non si modifica più (entro una certa approssimazione). La convergenza è sicura se ci si trova nelle condizioni di applicare un particolare teorema che stabilisce una condizione sufficiente ma non necessaria per la convergenza dei metodi iterativi. Tralasciamo il teorema completo, ma diciamo almeno che per le funzioni a una sola incognita – come la nostra – è sufficiente che la derivata prima nel punto dell'intersezione sia positiva e minore di uno; se non lo è, basta invertire y con x.

Come esempio di prova, spieghiamo dettagliatamente come configurare *geoCalc* per risolvere il nostro problema. Nella casella B6 scriviamo il valore numerico 1 (questo è il valore iniziale che assegnamo a x). Nella casella B5 immettiamo la formula:

= SQRT(LN(B6) + B6*2)

Istantaneamente il foglio ne calcola il risultato visualizzando 1,41421356 (accuratezza di 9 cifre) nella casella B6. A questo punto, per creare il riferimento circolare che produce l'iterazione (un riferimento circolare si crea quando la formula relativa a una casella indirizza il valore di un'altra casella che a sua volta contiene una formula che indirizza la prima

casella), inseriamo in B6 la semplice formula:

= B5

Il foglio inizia a eseguire una serie di calcoli iterativi che producono una sequenza di risultati che convergono verso la soluzione dell'equazione. *GeoCalc* blocca l'iterazione quando l'accuratezza di nove cifre con cui esegue i calcoli non è più sufficiente per rilevare una differenza fra l'ultimo valore di B5 e il nuovo valore calcolato. Il risultato della nostra equazione è 2,36393887. L'errore commesso da *geoCalc* nella determinazione della soluzione è dell'ordine di 10^{-8}. In questo modo abbiamo sfruttato la particolare struttura "in parallelo" del foglio per risolvere un'equazione trascendente tramite il metodo iterativo.

Possiamo ora fare alcune considerazioni di sicuro interesse per gli utenti di *geoCalc*. Innanzitutto, se *geoCalc* non giunge a una divergenza inferiore a 10^{-8} entro 300 iterazioni, blocca automaticamente il ricalcolo del foglio. Nel nostro caso, la funzione prescelta converge molto rapidamente, e quindi sono necessarie poche iterazioni. Ma in altri casi la scelta di un valore iniziale inadeguato può allungare considerevolmente i tempi. La seconda considerazione mette in luce una deficienza di *geoCalc*, irrilevante nelle applicazioni consuete ma decisamente fastidiosa nelle applicazioni che conducono a riferimenti circolari, come quella appena descritta. Il problema risiede nei valori iniziali, che in una struttura parallela iterativa non possono appartenere a caselle che non rientrano nell'iterazione.

Il valore iniziale di x costituisce ovviamente un'ipotesi di partenza del processo: in qualche modo, questo dato iniziale dev'essere reso noto al foglio elettronico nella prima fase di calcolo (la prima iterazione). Dal momento che questa formula assume come valore di x il contenuto della casella B6, il valore iniziale dev'essere memorizzato in quella casella. Ma perché l'iterazione cominci, la stessa casella deve anche contenere la formula = B5 che crea il riferimento circolare. Ecco allora che nasce l'esigenza di attribuire alla stessa casella una formula e un valore iniziale. Dovrebbe essere possibile congelare il foglio elettronico, configurarlo, immettere il valore iniziale, e poi scongelarlo. Purtroppo questa operazione non è consentita da *geoCalc*, e quindi occorre studiare un modo per scavalcare il problema.

Una soluzione consiste nel memorizzare in B6 il valore iniziale di x, e successivamente memorizzarvi la funzione = B5. *GeoCalc*, quando calcola la funzione relativa a B5 sfrutterà il valore di B5 (1 nel nostro caso), mentre nelle successive iterazioni continuerà a utilizzare l'ultimo valore calcolato. Quando l'iterazione è giunta a una soluzione, per ripartire con un altro valore iniziale di x occorre inserire nuovamente tale valore in B6, cancellando la formula ivi contenuta, e poi riscrivere ancora la formula perché l'iterazione inizi. Comunque, a parte questo fastidioso problema, *geoCalc* ci ha aiutati nella soluzione iterativa di un'equazione, mettendo in luce la facilità d'impiego del foglio elettronico rispetto a qualunque linguaggio di programmazione: non abbiamo dovuto stilare nessun algoritmo, ma solo creare una relazione strettamente legata alla formulazione matematica del problema.

Una singolare applicazione

Per concludere questa sommaria panoramica sugli usi non prettamente finanziari del foglio elettronico, vediamo come farlo diventare addirittura un "terreno di coltura" per un automa cellulare. Avvertiamo subito che con *geoCalc* questo tipo di simulazione non è possibile, se non seguendo vie particolarmente tortuose per superare alcuni limiti intrinseci del programma.

D'altra parte di fogli elettronici ce ne sono altri, e ci sembra interessante proporne un impiego piuttosto inconsueto come l'automa cellulare, ovvero un insieme di caselle che seguono un comportamento prestabilito. Per non complicare troppo il discorso, riduciamo la trattazione agli automi cellulari uniformi bidimensionali, nei quali ogni cellula segue regole di comportamento uguali a tutte le altre. Un esempio semplicissimo potrebbe essere un ambiente cellulare nel quale ogni cellula nasce e sopravvive solo se è circondata nelle quattro direzioni principali da almeno due cellule "vive". Non è difficile rendersi conto che un foglio elettronico è l'ideale per una rappresentazione di questo genere, dal momento che possiede già una struttura a caselle. Per fissare le idee, stabiliamo che una casella viva contenga il valore 1 e una casella morta il valore 0.

Il tipo di automa cellulare che proponiamo è simile al famoso gioco *Vita*, nato oltre dieci anni fa negli Stati Uniti: si prendono in considerazione le otto cellule che nelle otto direzioni circondano la generica cellula, e si stabiliscono alcune regole di sopravvivenza. Quelle che proponiamo noi sono queste: una cellula muore di solitudine se non è circondata da almeno quattro cellule vive; nasce se è circondata da più di tre cellule vive; muore per asfissia se è circondata da più di sei cellule vive. Per costruire l'automa si deve individuare la formula che descrive il comportamento della generica cellula, e associarla a tutte le caselle del foglio. Con *geoCalc* la formula da inserire nella casella B2 sarebbe: = INT(INT(Somma/3) * INT(1/(INT(Somma/7) + 1))), nella quale Somma costituisce la somma dei valori contenuti nelle otto caselle adiacenti, e la funzione INT tronca la parte decimale dell'argomento. Somma, sempre con *geoCalc* si esprimerebbe come: SUM (A1:C1,A3:C3,A2,C2). Le caselle adiacenti a quella in esame possono assumere soltanto il valore 1 o il valore 0, e quindi la variabile Somma risulta compresa fra 0 e 8. La formula può apparire un po' complicata, ma questo dipende dal fatto che *geoCalc* non prevede operatori logici o booleani. Gli indirizzi che compaiono nella formula non sono preceduti dal simbolo $, e quindi sono relativi: questo consente di copiarla senza ulteriori modifiche in tutte le caselle del blocco selezionato, e dar vita all'automa cellulare. Con *geoCalc*, come già accennato, è impossibile definire i valori iniziali e questo è il limite principale del programma, almeno per le applicazioni matematiche. Ma *geoCalc* non è fortunatamente l'unico foglio elettronico di cui è possibile avvalersi.

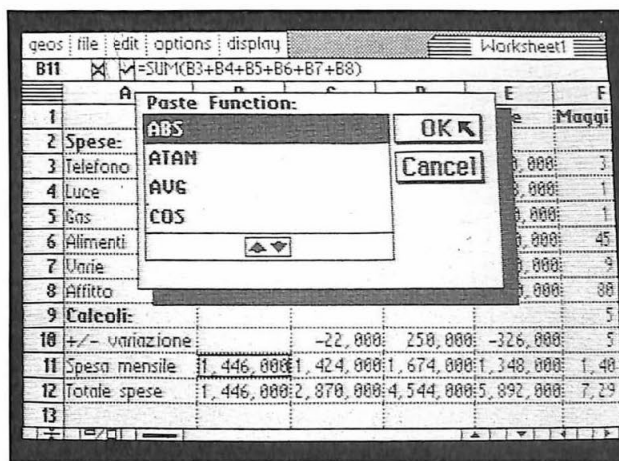
Per chi voglia fare altri esperimenti dello stesso tipo, riportiamo anche la legge del famoso automa cellulare *Vita*, ideato da John Horton Conway. Le regole sono: morte per isolamento con una o nessuna cellula circostante; sopravvivenza con due cellule circostanti; nascita con tre cellule circostanti; morte per asfissia con più di tre cellule circostanti. Speriamo con questi spunti un po' curiosi di aver stimolato la fantasia di chi ama la sperimentazione, e magari ha già da tempo messo il proprio foglio elettronico nel cassetto. In realtà questi programmi sono strumenti la cui efficacia poggia proprio sulla semplicità, e risultano spesso insostituibili sia nello studio che nel lavoro. E, come abbiamo visto, possono diventare anche divertenti passatempi per le serate invernali. **L.G.**

nei calcoli, ma di cui non si conosce ancora il valore.

Alcune funzioni, come per esempio SUM (somma un insieme di valori), permettono un numero variabile di parametri. In questi casi, *geoCalc* consente d'indicare i parametri sotto forma d'insiemi e unioni. Un insieme è un gruppo di caselle identificabile dalle due caselle d'angolo (per esempio, A1:C3 individua le caselle A1, A2, A3, B1, B2, B3, C1, C2, C3). Un'unione è una serie di caselle disgiunte (per esempio, A1, B25, F12).

Le funzioni matematiche offerte da *geoCalc* sono molte, come si è visto, ma è un peccato che manchino completamente gli operatori logici (AND, OR, <, >, <>...) e le strutture di controllo (IF, THEN, ELSE), la cui presenza è consueta in tutti i fogli elettronici. Si tratta di una grave mancanza, che per essere scavalcata costringe a intraprendere strade tortuose, non sempre praticabili. Inoltre, non è possibile associare a una casella un algoritmo composto da più di una funzione; anche questo rappresenta un grande limite, che si evidenzia soprattutto quando si desidera associare a una sola casella complesse formulazioni logico-matematiche. Se non si ricorda una funzione è possibile consultare un prontuario d'aiuto, che *geoCalc* visualizza in un box di dialogo quando si attiva la voce Paste function del menu Options.

Le impressioni

GeoCalc è facile da usare. Il manuale è strutturato in modo da condurre l'utente a una piena conoscenza dell'applicazione attraverso un'adeguata serie di esperienze che potremmo definire un minicorso. Sul disco è presente un esempio diverso da quello del manuale, per illustrare ulteriormente le caratteristiche del programma. Sebbene il manuale sia in inglese e riproponga una volta di più lo stile logorroico della Berkeley, è sufficientemente chiaro, e soprattutto è ricco di esempi nelle fasi più delicate (arriva perfino a introdurre nozioni matematiche di base non direttamente correlate con il pacchetto, come la definizione di talvolta inediti, come il raggruppamento di caselle, ha alcuni difetti per nulla trascurabili, come la mancanza degli operatori logici e delle strutture di controllo, l'impossibilità di associare alla

Il box che permette la scelta delle funzioni matematiche

radiante e la conversione radianti/gradi, gradi/radianti).

Il programma, come c'era da aspettarsi, è in grado di comunicare con le altre applicazioni *GEOS* compatibili come *geoWrite* e *geoFile*, e di caricare i desk accessory presenti su disco.

Per quanto la Berkeley abbia fatto notevoli sforzi per rendere più veloci gli accessi al disco, la struttura overlay che per ragioni di memoria *GEOS* impone alle applicazione più complesse è sempre fonte di lunghe attese. Con un'espansione RAM come la 1764 (256K), la situazione migliora notevolmente. A proposito di espansione, ricordiamo che in ambiente *GEOS* l'espansione 1700 (128K) permette di accelerare il trasferimento dati e di eseguire il reboot del sistema dall'espansione. La 1764 (256K) aggiunge la possibilità di ottenere uno shadowed disk o un RAM disk. La 1750 (512K) permette di aggiungere contemporaneamente sia lo shadowed disk sia il RAM disk. Insieme al mouse, l'espansione RAM costituisce uno strumento di lavoro fondamentale, che consigliamo vivamente agli utenti di *GEOS*.

In conclusione, *geoCalc*, pur offrendo gadget interessanti e casella un algoritmo, ovvero una sequenza di operazioni da svolgere in un certo ordine. Un altro aspetto da non sottovalutare, in un ideale confronto con analoghi strumenti di lavoro, è la lentezza dello scroll, che qualche volta riesce davvero a essere esasperante. Dal punto di vista delle pure prestazioni di calcolo si rivela quindi inferiore a prodotti come *Calc Result* e *Multiplan*. Tuttavia, per gli utenti del C-64 e del C-128 quest'applicazione è abbastanza interessante, un foglio elettronico semplice e piacevole da consultare, in grado d'introdurre nel sempre più articolato mondo di *GEOS* quasi tutte le più utili caratteristiche degli spreadsheet. È consigliabile soprattutto per chi ha già altri pacchetti *GEOS*, e non vuole affrontare gli ostacoli iniziali che – in una certa misura – si devono sempre superare con applicazioni che adottano soluzioni di lavoro del tutto nuove. ■

I pacchetti GEOS sono disponibili presso:

Lago snc
(geoCalc, Lire 75.000)
Via Napoleona, 16
22100 Como
(tel. 031/300174)

GRAFICA AVANZATA CON LE ESPANSIONI RAM 1700 E 1750

Con questo programma, un C-128 e un'espansione di memoria, potrete ottenere effetti d'animazione 3D che stupiranno chiunque ha una certa familiarità con i computer a otto bit

di Louis R. Wallace

Nonostante abbiano fatto la loro comparsa già da diverso tempo, le espansioni di memoria RAM 1700 e 1750 non hanno ancora destato un'attenzione sufficiente per convincere le software house a realizzare programmi in grado di sfruttarne appieno le potenzialità.

La carenza di software, comunque, non deve indurre l'acquirente a considerare questi prodotti come oggetti di scarsa utilità: la potenza e la duttilità di queste cartucce (particolarmente evidenti nei comandi STASH, FETCH e SWAP del Basic), sono in grado di aumentare enormemente le prestazioni delle macchine a otto bit della Commodore. Le due cartucce differiscono esclusivamente per la quantità di memoria RAM che contengono. La 1700 è dotata di 128K di RAM, mentre la 1750 ne ha 512.

I comandi di base hanno tutti la stessa sintassi, dal momento che prevedono – nello stesso ordine – gli stessi parametri, ovvero il numero di byte da trasferire, l'inizio dell'area considerata nella RAM interna del C-128, il banco di memoria dell'espansione e l'indirizzo al suo interno dell'area di memoria RAM considerata. Queste informazioni vengono impiegate da ogni comando in modo diverso.

L'esatta sintassi è la seguente:

```
STASH  #bytes,intsa,expsa,
       exbno
FETCH  #bytes,intsa,expsa,
       exbno
SWAP   #bytes,intsa,expsa,
       exbno
```

In questo contesto, **intsa** rappresenta l'indirizzo di partenza nella memoria del C-128 (0-65535), **expsa** l'indirizzo di partenza nel banco dell'espansione, e **exbno** il numero del banco di memoria dell'espansione (0-1 per la 1700, e 0-7 per la 1750). **#bytes** naturalmente indica il numero di byte sui quali il comando deve agire (1-65536).

STASH trasferisce un'area di memoria dal C-128 all'espansione RAM. FETCH trasferisce un'area di memoria dall'espansione RAM al C-128. SWAP scambia un'area di memoria del C-128 con un'area

di memoria dell'espansione RAM.

Sapendo esattamente in quale punto della memoria interna del C-128 sono contenuti i nostri dati, diventa per esempio possibile il loro immediato trasferimento nella RAM esterna. La velocità di trasferimento è vicina ai 200K al secondo, e la si ottiene agendo esclusivamente da Basic!

Vi sono molte informazioni che possono essere immagazzinate nei banchi di memoria RAM da 64K dell'espansione installata. Se si dispone del software necessario per creare un RAM disk, l'espansione può essere utilizzata come disk drive virtuale, ottenendo così tempi di accesso ai file estremamente veloci. È possibile utilizzare questa particolare configurazione con i propri programmi per immagazzinare grafici, sprite, o figure che possono così essere riutilizzati rapidamente in ogni momento. Il programma che presentiamo, *3D Plot*, può essere considerato come un suggerimento per un'intera classe di possibili applicazioni.

Un'animazione 3D

Fra le varie applicazioni dei computer in campo grafico, l'animazione di figure tridimensionali è una delle più affascinanti, ma anche una delle più complicate. Con una macchina a otto bit come il C-128, i problemi sono principalmente la limitata disponibilità di memoria RAM e la lentezza della CPU. Grazie a un'espansione RAM questi problemi vengono attenuati, e diventa molto soddisfacente poter sfruttare le sue elevate prestazioni per realizzare applicazioni di questo tipo. I risultati che si ottengono sono straordinari: immagini tridimensionali che si muovono e ruotano con una precisione e una continuità tale da rendere inevitabile il paragone con le animazioni cinematografiche.

Il programma *3D Plot* è un esempio di come le espansioni RAM possono venire impiegate per produrre un'animazione 3D. Il programma genera una serie d'immagini in tre dimensioni di una superficie nello spazio. Una volta disegnata, ogni immagine viene trasferita nella RAM (e sul disco), quindi lo schermo si cancella e viene generata una nuova versione dell'immagine, leggermente diversa dalla precedente. Il ciclo si ripete per un certo numero di volte, fino ad avere una quantità d'immagini sufficiente per realizzare una sequenza animata. Le varie immagini vengono richiamate in sequenza dalla RAM esterna così rapidamente da dare l'impressione di un movimento in tempo reale molto ben definito e continuo. Naturalmente un simile procedimento implica tempi di generazione delle immagini molto lunghi e, sebbene il modo a 2 MHz del C-128 (che disabilita lo schermo a 40 colonne) consenta una maggiore rapidità, l'attesa risulta drasticamente lunga.

Comunque le varie immagini vengono salvate anche su disco, oltre che nell'espansione RAM, e non si deve quindi ripetere la stessa sequenza di operazioni ogni volta che si desidera visualizzare l'animazione. Prima di disegnare un'immagine, infatti, il programma controlla i dati contenuti sul disco; se fra questi è compresa l'immagine richiesta, la macchina provvede automaticamente a caricarla nell'espansione. In caso contrario l'immagine viene disegnata, registrata su disco e soltanto dopo queste due lente operazioni viene caricata nell'espansione. Il procedimento ora descritto permette quindi di arrestare l'esecuzione del programma in ogni momento, senza per questo rischiare di perdere le immagini già create in precedenza. Una volta registrata su disco, una sequenza può essere richiamata in memoria in tempi relativamente brevi.

Istruzioni per l'uso

Una volta mandato in esecuzione, il programma richiede all'utente d'indicare il tipo di configurazione disk drive/espansione che sta utilizzando. Se si possiede un 1541 o un'espansione RAM 1700, il numero d'immagini che *3D Plot* può salvare su disco e nell'espansione è solo 16. Se invece si dispone del 1571 e dell'espansione RAM 1750, *3D Plot* ha lo spazio sufficiente per salvare 32 immagini.

Dopo la sezione della configurazione, viene chiesto all'utente se desidera formattare il disco presente nel disk drive. La prima volta che ci si serve del programma è necessario infatti inserire

nel disk drive un disco vergine; *3D Plot* lo formatta, e ci salva sopra se stesso. Prima di formattare un nuovo disco il programma controlla se contiene già immagini, e in questo caso dà all'utente la possibilità d'interrompere la procedura. A questo punto, procede alla creazione delle immagini che non trova su disco, e salva tutte le immagini nell'espansione RAM. Quando anche questa fase è terminata, il programma entra in un loop che visualizza in sequenza tutte le immagini presenti nell'espansione RAM, a una velocità così elevata da rendere l'animazione particolarmente efficace.

Durante la creazione delle immagini in memoria, il programma abilita il modo fast del C-128. Non bisogna quindi angosciarsi troppo per i prolungati periodi d'inattività dello schermo: la generazione della sequenza animata richiede infatti circa tre ore.

Il programma è semplice, ma i risultati che si riescono a ottenere con le espansioni RAM sono davvero molto interessanti. L'animazione generata è una fra le migliori sequenze animate mai viste su un C-128.

Listato: *3D Plot*

```
10 REM 3D PLOT: ANIMAZIONI 3D
20 REM DI LOUIS R. WALLACE
30 REM
40 REM RICHIEDE L'ESPANSIONE RA
   M 1700/1750.
50 REM SCEGLIERE L'OPZIONE CORR
   ISPONDENTE
60 REM ALLA CONFIGURAZIONE ESPA
   NSIONE
70 REM RAM/DISK DRIVE CHE SI PO
   SSIEDE:
80 REM 1700 + 1541 SCEGLIERE 15
   41
90 REM 1700 + 1571 SCEGLIERE 15
   41
100 REM 1750 + 1541 SCEGLIERE 1
    541
110 REM 1750 + 1571 SCEGLIERE 1
    571
120 REM L'OPZIONE 1 IMPIEGA 128
    K SU DISCO/RAM.
130 REM L'OPZIONE 2 IMPIEGA 256
    K SU DISCO/RAM.
140 REM INSERIRE UN DISCO DA FO
    RMATTARE LA
150 REM PRIMA VOLTA CHE SI ESEG
    UE IL PRG.
160 REM RISPONDERE SI ALL'OPZIO
    NE DI
170 REM FORMATTAZIONE SOLO LA P
    RIMA
180 REM VOLTA.
190 :
200 GRAPHIC1,1:GRAPHIC0
210 TRAP 1360
220 :
230 SCNCLR
240 PRINT"                  3D PL
    OT"
250 PRINT
260 PRINT:PRINT"CONFIGURAZIONI
    POSSIBILI:"
270 PRINT:PRINT"1.  1541 - RAM
    1700 O RAM 1750"
275 PRINT"     1571 - RAM 1700"
280 PRINT:PRINT"2.  1571 - RAM
    1750"
290 PRINT:PRINT"SCEGLIERE LA PR
    OPRIA (1 O 2)"
300 GETKEY DN$:IF DN$<>"1" AND
    DN$<>"2" THEN 300
310 IF DN$="1" THEN NF=16:ELSE
    NF=32
320 PRINT:PRINT:PRINT"SONO NECE
    SSARI";NF*32;"BLOCCHI SU DI
    SCO"
330 PRINT"PER MEMORIZZARE LE IM
    MAGINI 3D"
340 PRINT:PRINT"FORMATTO IL DIS
    CO (S/N)?"
350 GETKEY FD$:IF FD$<>"S" AND
    FD$<>"N" THEN 350
360 IF FD$="S" THEN BEGIN
370 ::PRINT"INSERIRE IL DISCO D
    A FORMATTARE"
375 ::PRINT"E PREMERE RETURN"
380 ::GETKEY A$:IF A$<>CHR$(13)
    THEN 380
390 ::RENAME "FRAME 1" TO "FRAM
    E 1"
400 ::IF DS=63 THEN PRINT:PRINT
    "SUL DISCO SONO PRESENTI DE
    LLE IMMAGINI!":PRINT"CONTIN
    UO LA FORMATTAZIONE (S/N)?"
    :ELSE GOTO 440
410 ::GETKEY A$
420 ::IF A$<>"S" AND A$<>"N" TH
    EN 410
430 ::IF A$="N" THEN 450
440 ::HEADER"3D PICTURES", ITD
450 ::DSAVE"@3DPLOT"
460 BEND
470 PRINT:PRINT"PREMERE UN TAST
    O PER INIZIARE"
480 GETKEY A$:PRINT"PROCEDO..."
490 PH=.1:B=0:N=0
500 DIM YM(320),YX(320)
510 :
515 REM LOOP DI GENERAZIONE DEL
    LE IMMAGINI
520 FOR A=0 TO NF-1
530 RH=100:D=800:TH=.1
540 S1=SIN(TH):C1=COS(TH)
550 S2=SIN(PH):C2=COS(PH)
560 FORI=0 TO 320:YM(I)=199:YX(
    I)=0:NEXT
570 :
580 DEF FN F(W)=14*EXP(-.04*W)*
    COS(.15*W)
590 F=0:FL=0:Z=0:W=0:XE=0:YE=0:
    ZE=0
600 SX=0:SY=0:DX=0:SL=0:YP=0:XP
    =0
610 GRAPHIC1,1:FAST:OX=0:OY=0
620 NM$="FRAME"+STR$(A)
630 RENAME (NM$) TO (NM$):REM E
    SISTE?
640 MF=0:IF DS=63 THEN MF=1:GOT
    O680
650 :
660 GOSUB 990:REM NO, CREO L'IM
    MAGINE 3D
670 :
680 SLOW
690 PH=PH+.1
700 IF MF=1 THEN 730
710 BSAVE (NM$),B0,P8192 TO P16
    181
720 GOTO 740
730 BLOAD (NM$),B0,P8192
740 STASH 8000,8192,B,N
750 B=B+8000:CT=CT+1
760 IF CT=8 THEN CT=0:N=N+1:B=0
770 NEXT A
780 :
790 GRAPHIC0:SLOW
800 SCNCLR
810 GRAPHIC1,1
820 :
830 REM DOPPIO LOOP D'ANIMAZION
    E
840 B=0:N=0:CT=0
850 FOR FR=0 TO NF-1
860 FETCH 8000,8192,B,N:B=B+800
    0:CT=CT+1
870 IF CT=8 THEN CT=0:N=N+1:B=0
880 NEXT
890 :
900 B=56000:CT=0:N=1:IF DN$="2"
     THEN N=3
910 FOR FR=(NF-1) TO 0 STEP -1
920 FETCH 8000,8192,B,N:B=B-800
    0:CT=CT+1
930 IF CT=8 THEN CT=0:N=N-1:B=5
    6000
940 NEXT
950 :
960 GOTO 840
970 :
980 REM ALGORITMO DI DISEGNO 3D
990 FOR X=12 TO -12 STEP -1
1000 FL=0
1010 FOR Y=-12 TO 12 STEP .5
1020 W=X↑2+Y↑2
1030 Z=FN F(W)
1040 GOSUB 1090:REM PROSPETTIVA
1050 NEXT Y
1060 NEXT X
1070 RETURN
1080 :
1090 XE=-X*S1+Y*C1
1100 YE=-X*C1*C2-Y*S1*C2+Z*S2
1110 ZE=-X*S2*C1-Y*S1*S2-Z*C2+R
     H
1120 SX=D*XE/ZE+159
1130 SY=-5*D*YE/ZE/12+99
1140 :
1150 IF SX<1 OR SX>319 THEN 127
     0
1160 IF FL=0 THEN FL=1:F=0:GOTO
      1270
1170 DX=OX-SX:IF DX=0 THEN DX=1
1180 SL=(OY-SY)/DX
1190 YP=OY
1200 :
1210 FOR XP=INT(OX)+1 TO SX
1220 YP=YP+SL
1230 IF YP<=YM(XP) THEN 1280
1240 IF YP>=YX(XP) THEN 1320
1250 F=0
1260 NEXT XP
1270 OX=SX:OY=SY:RETURN
1280 YM(XP)=YP
1290 IF F=0 AND YP>0 THEN LOCAT
     E XP,YP:F=1
1300 IF(XP<0ORXP>319)OR (YP<0 O
     R YP>199) THEN 1260:ELSE D
     RAW TO XP,YP
1310 GOTO 1260
1320 YX(XP)=YP
1330 IF F=0 AND YP>0 THEN LOCAT
     E XP,YP:F=1
1340 IF(XP<0ORXP>319)OR (YP<0 O
     R YP>199) THEN 1260:ELSE D
     RAW TO XP,YP
1350 GOTO 1260
1360 GRAPHIC0:SLOW
```

CORSO DI PROGRAMMAZIONE IN C PER L'AMIGA: PRIMA PUNTATA

INTRODUZIONE ALLA PROGRAMMAZIONE IN LINGUAGGIO C

Inizia con questo articolo un corso che porterà alla perfetta padronanza del linguaggio C per l'Amiga. Gradino per gradino viene descritto il flusso di controllo di un programma in C: la creazione, la compilazione, il debug, l'uso delle librerie

di Eugene P. Mortimore

Questo articolo è il primo di una serie che intende insegnare il linguaggio C, concentrandosi in particolare sulla programmazione dell'Amiga. Per come è stato strutturato, il corso non vuole sostituirsi a un testo didattico sul linguaggio C. L'obiettivo è infatti ben lontano da una trattazione teorica senza riferimenti all'ambiente di lavoro di cui si dispone. Al contrario, il corso cercherà di mostrare le potenzialità offerte dal binomio Amiga/C, cioè sarà centrato sull'apprendimento di questo linguaggio direttamente sul campo, a stretto contatto con tutte le potenzialità dell'Amiga che in C possono essere ampiamente sfruttate, molto più che in Basic o in Pascal. Il programmatore che non vuole scendere a livelli troppo bassi (Assembly per intenderci), tramite il linguaggio C riesce a mantenere un controllo di basso livello sulle strutture di dati dell'Amiga, sfruttando un linguaggio molto meno ostico dell'Assembly.

Il corso è quindi rivolto principalmente a coloro che desiderano apprendere il linguaggio C e impiegarlo con il proprio Amiga. Gli argomenti che verranno trattati costituiranno una buona base di appoggio per chi ha già programmato in Basic o in Pascal favorendone il passaggio al C, ma catalizzeranno l'attenzione anche dei lettori che già conoscono il C ma non conoscono l'Amiga, oppure che conoscendo sia il C sia l'Amiga sono sempre alla ricerca di spunti e idee per migliorare le proprie capacità. Proprio per la sua unicità, questo corso pretende d'insegnare il C con l'Amiga e per l'Amiga, e come tale rappresenta una fondamentale fonte d'informazioni per completare i libri di programmazione sul C oggi disponibili. Tra le cose che infatti dobbiamo dare per scontate, la principale è che ogni lettore disponga almeno di un buon libro dedicato alla programmazione in linguaggio C.

Lo scopo principale di questo primo articolo è di preparare il terreno per gli articoli successivi, fornendo tutte le necessarie conoscenze di base. Al termine di questa prima lezione verranno illustrati due semplici esempi di programmi in C che hanno soltanto la funzione di mostrare in che modo viene strutturato il flusso di controllo di un generico programma in C. Verranno infine elencati i passi necessari per trasformare questi due esempi in programmi eseguibili, senza però entrare in dettagli che costituiranno gli argomenti delle prossime puntate.

La programmazione in C dell'Amiga

Programmare in C con l'Amiga richiede la costruzione di un particolare schema mentale: una filosofia della programmazione che si può riassumere in poche parole: «Trarre il massimo vantaggio da quello che è già stato fatto». Il sistema operativo dell'Amiga è stato progettato proprio per permettere ai programmatori di trarre il massimo vantaggio dagli speciali chip custom della macchina. Le centinaia di routine che il sistema mette a disposizione servono proprio a rendere la vita più facile ai programmatori, che riescono così a sfruttare ogni risorsa in modo relativamente semplice.

In questa impresa, un sistema di convenzioni era necessario, non solo per progettare il sistema, ma anche per aiutare i programmatori nella creazione del software. Questo sistema di convenzioni è contenuto nei file INCLUDE, inesauribili fonti di sapere sulle scritture simboliche adottate per rappresentare i dati da gestire (questi file verranno

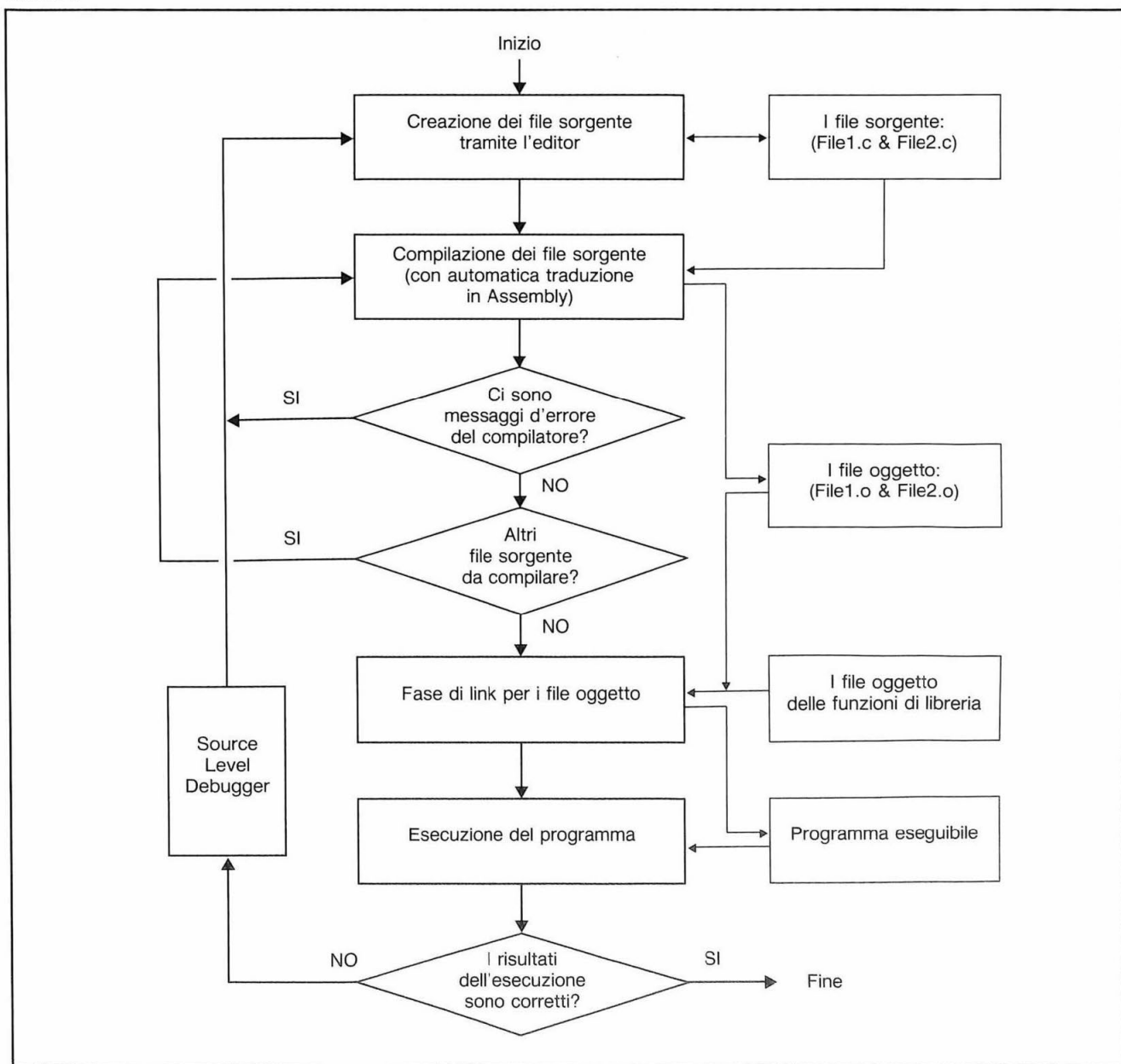
meglio descritti più avanti).

Il risultato di questi sforzi è stato un hardware di altissimo livello, e un software sistema estremamente complesso, ricco di routine che i programmatori possono impiegare per accedere alle speciali caratteristiche della macchina. Proprio per questo, le prime energie dell'aspirante programmatore devono essere spese per studiare con attenzione i risultati complessivi di questo sforzo gigantesco. Solo così riuscirà a capire fino in fondo che la parte più complessa del lavoro di programmazione, in realtà, è già stata fatta ed è a disposizione di tutti.

Tutte le funzioni di sistema sono state raggruppate in raccolte di routine, le cosiddette librerie dell'Amiga. La maggior parte di queste librerie sono state installate nei 256K del chip ROM, e le rimanenti sono disponibili sul disco sistema. La differenza tra le librerie di funzioni presenti nel ROM Kernel e quelle su disco è praticamente nulla per il programmatore, dal momento che è il sistema a curarsi di mantenere in memoria i codici richiesti, eventualmente liberando memoria occupata da altre librerie che non servono più.

Dal sorgente al programma eseguibile

La Tavola 1 illustra in generale la procedura da seguire per programmare in linguaggio C. Si articola nei cinque seguenti gradini:
– progettare e scrivere i file sorgente in linguaggio C che rappresentano il programma. Ogni file sorgente contiene la rappresentazione in linguaggio C di un modulo del programma, e deve avere

Tavola 1

l'estensione ".c" nel proprio nome
– compilare individualmente ciascun file sorgente per produrre l'associato file oggetto; ogni file oggetto prodotto dal compilatore avrà automaticamente l'estensione ".o" nel nome
– sottoporre alla fase di link ciascuno dei file oggetto che definiscono il programma completo, e i file oggetto richiesti dal sistema di programmazione proprio del linguaggio C (le librerie linked, che discuteremo più avanti nell'articolo). È in questa fase che si ottiene, salvo errori, il programma eseguibile (si noti che il nome del file contenente il programma eseguibile non ha alcuna estensione obbligatoria)
– mandare in esecuzione il programma per mettere in evidenza eventuali errori, e infine
– eseguire il debug per correggerli.

Questa è in sostanza la prassi da seguire per realizzare un programma in C. Come si può vedere, dobbiamo servirci di strumenti di lavoro come l'editor, il compilatore, il linker. Analizziamo ora cosa avviene durante ogni fase e quali caratteristiche presentano, o dovrebbero presentare, questi strumenti di programmazione.

La pianificazione e la scelta dell'editor

Prima di mettersi effettivamente al lavoro con l'editor si deve pianificare e progettare da un punto di vista concettuale le funzioni in linguaggio C che comporranno il nostro programma. In questa fase si decide di quali funzioni e variabili si servirà il programma, in che modo le informazioni verranno rappresentate e passate alle varie funzioni (ci sono due possibilità: che le informazioni vengano considerate argomenti oppure variabili esterne), e in generale come sarà organizzato il programma. Si deve inoltre progettare l'interfaccia utente del programma, ovvero i metodi per inserire i dati: schermi, menu, finestre e così via. Il processo di scrittura vero e proprio sarà enormemente semplificato se questa progettazione preliminare verrà effettuata in modo accurato e completo.

Successivamente si dovrebbe ricorrere a un editor abbastanza veloce che faciliti la scrittura del codice sorgente. Per gli scopi del programmatore la dote principale di un buon editor è la capacità di produrre velocemente e accuratamente codici sorgente di facile lettura. In poche parole, dovrebbe essere in grado di gestire più file aperti simultaneamente su più finestre, dovrebbe offrire operazioni di ricerca e sostituzione molto rapide e la capacità di creare rientranze diversificate per specifici blocchi di testo, in modo da facilitare la stesura di un testo sorgente di buona leggibilità.

La compilazione

Una volta creati i file sorgente con l'editor, il passo successivo è la compilazione; ciascun file del programma dev'essere compilato separatamente. Il compilatore C riceve un modulo sorgente in C e lo trasforma in un nuovo modulo sorgente in Assembly. Il risultato è un file di testo sorgente contenente la traduzione in Assembly dell'originale programma in C. Durante questa trasformazione, il compilatore C verifica implicitamente la corretta sintassi del codice sorgente, controlla che tutte le variabili abbiano riferimenti coerenti in tutti i file – sia in quelli del programma sia nei file INCLUDE associati – e quindi genera il file sorgente in Assembly, pronto per essere ceduto all'assembler (o compilatore Assembly). A scelta, è possibile effettuare anche solo il controllo sintattico di un programma. Questa operazione generalmente preliminare può essere eseguita con un programma indipendente di controllo sintattico, o con lo stesso compilatore (se lo prevede).

Se il compilatore scopre che gli mancano informazioni (per esempio ci sono riferimenti a variabili non definite), oppure che nel sorgente è presente un errore di sintassi, genera messaggi d'errore dai quali in genere si riesce a capire come si deve modificare il codice sorgente per renderlo compilabile. Un errore potrebbe per esempio essere il seguente: supponiamo che il programma dichiari una variabile denominata myTask come puntatore a una struttura di tipo Task. La composizione di questo tipo di struttura è definita nel file INCLUDE tasks.h. Quando il compilatore scorrendo il codice sorgente incontra la definizione della variabile myTask, va a consultare i file INCLUDE richiamati all'interno del sorgente per individuare la definizione della struttura Task. Se però il codice sorgente non richiama il file INCLUDE tasks.h tramite un comando #include, il compilatore non trova il riferimento desiderato, genera un messaggio di "errore fatale" e arresta la compilazione del programma. È in questo modo che il compilatore si assicura che tutti i riferimenti presenti all'interno del programma siano coerenti.

Quando il codice sorgente è stato corretto, si prova a compilarlo nuovamente. Se il compilatore C accetta il codice sorgente – cosa che garantisce la correttezza della sua sintassi e la coerenza dei riferimenti – provvede a compilarlo generando il relativo file sorgente in Assembly. Se non viene diversamente istruito, procede poi automaticamente a chiamare l'assembler. Questo secondo compilatore riceve il file sorgente in Assembly e lo traduce in linguaggio Assembly del 68000, cioè in linguaggio macchina. Il terzo file che così si ottiene non è più un sorgente, ma un oggetto. Con questa seconda denominazione si vuole indicare che il file ha già le sembianze del programma finale, è composto da istruzioni in linguaggio macchina e dati (e quindi non è consultabile come un file di testo), ed è pronto per essere sottoposto alla fase di link.

Nel caso del pacchetto compilatore/assembler/linker della Manx, salvo controindicazioni la compilazione effettuata dall'assembler è automatica, e il passaggio intermedio attraverso il file

sorgente in Assembly non è visibile, in quanto questo file viene automaticamente rimosso. Il programmatore chiama semplicemente il compilatore C e ottiene il file oggetto. Ovviamente se il programmatore desidera soltanto assemblare un programma in linguaggio Assembly, può servirsi dell'assembler indipendentemente dal compilatore C.

Quando tutti i moduli sorgente del programma sono stati compilati (con questa espressione si sottintende anche la loro trasformazione in codice Assembly oggetto), si procede alla fase di link. Bisogna mandare in esecuzione il linker specificando l'insieme complessivo dei moduli oggetto che costituiscono il programma, come indicato nella Tavola 1.

Il link e l'esecuzione del programma

Il linker provvede a mettere insieme i file oggetto che costituiscono i moduli del programma e i file oggetto precompilati delle librerie interne a cui il programma ricorre per funzionare nel sistema Amiga (le librerie linked). Il risultato prodotto dal linker è il programma eseguibile, l'obiettivo dell'intera procedura. Generalmente parlando, alcune funzioni del programma (o anche tutte) si riferiranno a variabili che non sono definite nelle funzioni stesse. Si tratta delle cosiddette variabili esterne: il programma vi fa riferimento, ma non contiene la loro definizione. Però, in qualche punto dell'insieme complessivo dei file oggetto, queste variabili esterne devono essere definite, altrimenti il linker non saprà come gestirle. Lo scopo del linker è cercare in quali file oggetto, tra tutti quelli coinvolti nel processo di link, sono definite le variabili. L'operazione ha successo se tutte le variabili esterne dichiarate nei moduli del programma risultano in qualche punto definite. Torneremo ancora su questo argomento nei prossimi articoli.

Oltre a risolvere i riferimenti a variabili esterne, il linker controlla anche che tutte le funzioni a cui si fa riferimento in ciascun modulo e che non sono definite nel modulo stesso, siano definite almeno in uno degli altri moduli che intervengono nella fase di compilazione. Se non trova la definizione di una funzione in nessuno dei moduli, accede ai file oggetto precompilati, assumendo che la particolare routine può essere o una routine standard del linguaggio C (per esempio, printf() o scanf()), o una routine caratteristica dell'ambiente Amiga. I file m.lib, mx.lib, ma.lib, o m8.lib sono esempi di librerie linked che, nel sistema della Manx, vengono consultate durante la fase di link quando nei moduli oggetto del programma s'incontra un riferimento a una routine matematica dell'Amiga. Questi file contengono routine matematiche precompilate, destinate specificamente al Motorola 68000. Anche questo argomento sarà ripreso in una delle prossime puntate del corso.

Se il linker produce un messaggio d'errore, può trattarsi di un errore nel codice sorgente in linguaggio C del programma, che dev'essere quindi corretto, o di un errore nella chiamata al linker. Il processo di link può infatti non funzionare correttamente se i nomi dei file oggetto sono sbagliati oppure se è sbagliato l'ordine in cui sono stati elencati. Quando tutti gli eventuali errori di questo tipo sono stati corretti, e né il compilatore né il linker danno luogo a messaggi d'errore, si ottiene finalmente un file eseguibile. È interessante notare che a questo punto il file, pur essendo eseguibile, non è detto che funzioni correttamente, dal momento che l'aver superato la fase di compilazione non può garantire la correttezza degli algoritmi impiegati.

A questo punto si tenta ovviamente di mandare in esecuzione il proprio programma. Se si tratta di un programma particolarmente breve e semplice, è molto probabile che venga subito eseguito correttamente... ma potrebbe anche non essere così. In questo contesto, "esecuzione corretta" significa che il programma fa quello che ci aspettiamo, e non produce risultati incompleti, sbagliati o diversi da quelli che ci interessano. Se invece si tratta di un programma di una certa complessità, che comprende più di un file sorgente e quindi più di un file oggetto, è probabile che non funzioni correttamente la prima volta che lo si manda in esecuzione. Può limitarsi a raggiungere risultati errati o addirittura causare il crash del sistema. In questo secondo caso dà come risultato un confuso messaggio di Guru Meditation, costringendo il programmatore a riattivare il sistema per poter ricominciare a lavorare.

Il debug

Dal momento che eseguire la riattivazione della macchina è uno spreco di tempo e di dati, un debugger (e soprattutto un source level debugger) può essere un aiuto prezioso per capire che cosa ha costretto il programma a interrompere la sua esecuzione oppure a dare risultati diversi dalle aspettative. Un source level debugger rappresenta quindi il successivo passo logico nella procedura di programmazione. Grazie a esso, il programmatore può mandare in esecuzione i vari elementi del codice sorgente uno alla volta e vedere i risultati dell'esecuzione passo dopo passo in modo da riuscire a identificare gli errori. Non si può fare scorrere il programma all'indietro come il nastro di un registratore, ma si può farlo scorrere in avanti "fotogramma dopo fotogramma", tenendolo sempre d'occhio grazie all'ausilio del source level debugger che permette di visualizzare sullo schermo, quando lo si desidera, i risultati ottenuti dal programma fino a quel momento. Inoltre, mentre si manda in esecuzione il programma, si possono impostare alcuni breakpoint per osservare come stanno andando le cose: fissare cioè alcuni punti all'interno del codice sorgente raggiunti i quali il sistema arresta automaticamente lo svolgi-

mento del programma per permettere di controllare i risultati raggiunti.

Questa e altre caratteristiche fanno del source level debugger uno strumento prezioso nella programmazione dell'Amiga. A questo scopo è particolarmente adatto l'omonimo programma *Source Level Debugger* della Manx. In uno dei prossimi articoli della serie descriveremo nei dettagli in che modo servircene.

Come rendere la programmazione più efficiente

Oltre all'enorme miglioramento nell'efficienza di programmazione dato dal source level debugger, è importante notare che l'Amiga offre altri vantaggi ai programmatori. Consente per esempio di trasferire su un RAM disk tutti i file relativi alla programmazione, in modo da rendere gli accessi particolarmente veloci. Tra i file da trasferire nel RAM disk vi sono gli strumenti base della programmazione: il compilatore, l'assembler e il linker. Bisognerebbe trasferirvi anche tutti i file sorgente e i file oggetto prodotti dal compilatore, e tutti i file INCLUDE relativi al programma. Se vi è abbastanza memoria disponibile, si può anche creare un RAM disk abbastanza grande da contenere tutti i file INCLUDE dell'Amiga. Basta creare un file batch di copia da eseguire quando la macchina ha completato la propria attivazione.

In generale, dal momento che tutti questi file di supporto uniti a un programma di medie dimensioni occupano circa un megabyte di memoria, è necessario un RAM disk pari o superiore a un megabyte. In altre parole, bisogna avere una macchina dotata di almeno due megabyte di memoria per poter creare un tale RAM disk e gestire tutte le altre operazioni con la massima rapidità. Se invece si ha a disposizione una quantità minore di memoria, si possono collocare nel RAM disk soltanto alcuni file opportunamente selezionati – per esempio, soltanto il compilatore, l'assembler, il linker e una libreria linked – e lasciare gli altri, come per esempio i file INCLUDE, sull'hard disk o sui dischi, anche se in questo modo le operazioni vengono inevitabilmente rallentate.

Le caratteristiche multitasking

Al secondo posto, come mezzo per rendere più veloce il lavoro di programmazione, vengono le caratteristiche multitasking dell'Amiga: configurando correttamente il proprio sistema si può procedere con la fase di compilazione e di link su uno dei propri file sorgente (processo in background) mentre se ne sta digitando un altro con l'editor (processo in foreground).

Il sistema operativo AmigaDOS è perfettamente in grado di gestire questa procedura. Si può così continuare a utilizzare il proprio editor per caricare, scrivere e salvare un file sorgente, mentre contemporaneamente il sistema compila uno o più file sorgente utilizzando i processi in background del CLI. Non ha nessuna importanza che ambedue i processi, quello in foreground e quello in background, debbano alternativamente accedere al disco (che sia un disco o un hard disk o un RAM disk). Il sistema multitasking dell'Amiga è abbastanza previdente da assicurarsi che venga mantenuta l'integrità di tutti i file – i file sorgente, i file oggetto e quelli eseguibili – nel corso di questa complessa procedura.

Nel caso del programma d'esempio che accompagna questa prima lezione, si può scrivere il primo file sorgente (File1.c), salvarlo e avviare un processo in background per compilarlo, e contemporaneamente procedere a digitare il secondo file sorgente (File2.c) utilizzando ancora l'editor. Quando è terminato il processo di stesura anche per File2.c, si può dare il via per la seconda volta alla compilazione ancora come processo in background (questo è possibile anche se l'operazione in background di compilazione di File1.c non si è ancora conclusa).

Se si hanno più di due file sorgente si può continuare per quanto è necessario, utilizzando quindi il sistema multitasking della macchina per velocizzare al massimo le fasi più ripetitive della programmazione e per eliminare i tempi morti. La sola restrizione a questa procedura è che non si può correggere con l'editor lo stesso file sorgente che è in fase di compilazione.

Le informazioni per la compilazione

La Tavola 2 illustra l'origine delle informazioni necessarie per trasformare il codice sorgente in codice oggetto e poi eseguibile. Lo scopo di questo diagramma è mostrare come il compilatore ottiene tutte le informazioni di cui ha bisogno per svolgere il suo compito.

Un programma è composto da uno o più file sorgente. Vedremo che ciascun singolo file sorgente di un programma richiama alcuni file INCLUDE dell'Amiga, e si riferisce alle definizioni di strutture e di variabili in essi presenti. Tutte queste informazioni si trovano nei file INCLUDE dei dispositivi e delle librerie. La Tavola 2 mostra inoltre i file INCLUDE specifici del programma, nei quali trovano posto alcune delle informazioni relative al programma (o anche tutte).

Si ricordi sempre, inoltre, che i programmi per l'Amiga si suddividono in due categorie fondamentali: quelli in linguaggio C standard, e quelli che aggiungono specifiche chiamate a funzioni dell'Amiga. I primi possono normalmente essere compilati con qualsiasi macchina, dal momento che contengono solo chiamate alle funzioni standard del C. I secondi, invece, sono per definizione quelli che fanno ricorso alle speciali caratteristiche hardware/software dell'Amiga, e quindi richiedono le informazioni contenute nei file INCLUDE dell'Amiga.

Lo scopo della Tavola 2 è di mostrare come le informazioni contenute nei file sorgente – dichia-

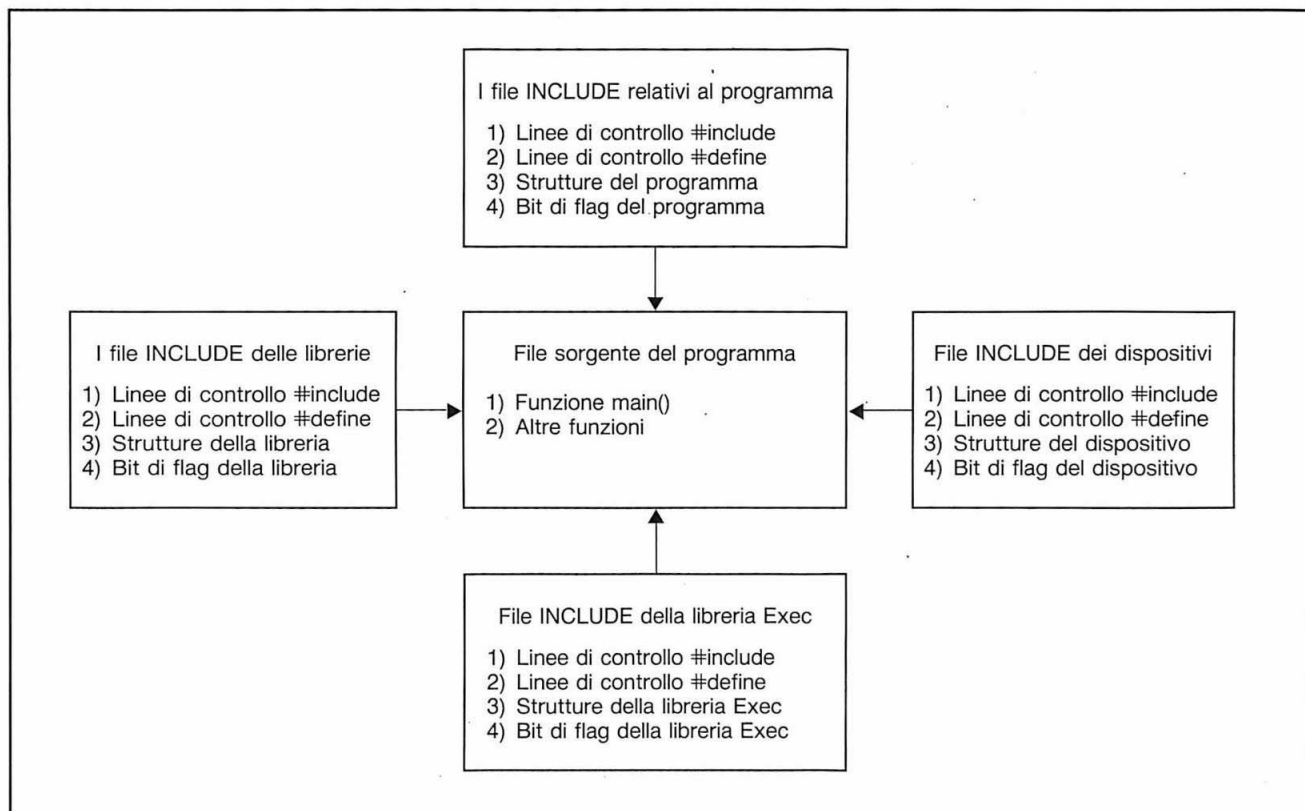



Tavola 2

razione di variabili, definizioni e dichiarazioni di strutture... – interagiscono con le informazioni contenute nei file INCLUDE della libreria Exec, con i file INCLUDE delle altre librerie e con i file INCLUDE dei dispositivi. Vediamo quali sono le informazioni necessarie al compilatore nella maggior parte dei casi.

Il linguaggio C standard

Un programma in linguaggio C standard, che non faccia uso delle funzioni specifiche dell'Amiga, potrebbe non aprire né librerie generiche né librerie dei dispositivi, e neanche servirsi della libreria Exec direttamente. In questo caso, i due grandi rettangoli che si vedono sulla sinistra e sulla destra della Tavola 2 non ci sarebbero. Nel caso più generale, comunque, ogni programma destinato all'Amiga interagisce con le routine dell'Exec, e come minimo deve contenere alcuni comandi #include che facciano riferimento ai file INCLUDE dell'Exec. Infatti questi particolari file contengono informazioni vitali per quasi tutti i programmi (per una più approfondita discussione della struttura ExecBase, si veda l'articolo "La struttura ExecBase dell'Amiga" in questo stesso numero), come per esempio la definizione delle variabili che definiscono quanti byte sono allocati in RAM per ogni tipo di variabile, presenti nel file INCLUDE types.h dell'Exec.

Inoltre, se il programma utilizza almeno una delle routine interne delle altre librerie dell'Amiga (oltre a quelle della libreria Exec), per esempio quelle della libreria Graphics o della libreria Intuition, il programma deve contenere l'opportuna linea di controllo che fa riferimento ai file INCLUDE specifici di quella libreria; ancora una volta viene usato un comando #include per mettere l'informazione a disposizione del compilatore. La necessità di ricorrere ai comandi #include vale infine anche per le librerie dei dispositivi, se il programma vi accede.

Nel suo complesso, il programma è quindi definito dalle informazioni contenute nei suoi file sorgente, compresi tutti i file INCLUDE richiamati, che diventano parte integrante del programma. In questo modo il compilatore dispone di tutti gli strumenti necessari per svolgere il suo compito. Per esempio, se il programma che intendiamo compilare definisce un altro task, e intende realizzare comunicazioni intertask, ha bisogno di creare un riferimento alla struttura Task dell'Exec, dichiarando una variabile che rappresenti la particolare struttura di tipo Task da aggiungere al sistema. Successivamente, per comunicare con il nuovo task, il nostro programma ha bisogno di creare message port e allocare segnali. Per far questo deve necessariamente dichiarare dati utilizzando i tipi previsti dal sistema e definiti nei file INCLUDE, come le strutture di tipo MessagePort e Message. Quando sono state definite le appropriate message port e i segnali, il programma può scambiare informazioni con il nuovo task facendo uso delle funzioni di gestione dei segnali e delle

message port, come AddPort, RemPort, AllocSignal e altre funzioni conservate in ROM nella libreria Exec. Per risolvere tutti questi riferimenti a tipi di variabili e tipi di funzioni, il compilatore si serve dei file INCLUDE che sono stati citati nel programma. I file INCLUDE della libreria Exec sono rappresentati dal largo rettangolo nella parte bassa della Tavola 2.

La descrizione non cambia quando si passa ai file INCLUDE delle altre librerie dell'Amiga, rappresentati nella parte sinistra della Tavola 2. Per esempio, se il programma intende aprire schermi o finestre, deve aprire la libreria Intuition (quindi è necessaria una chiamata alla funzione OpenLibrary dell'Exec) e utilizzare per le proprie strutture i tipi di dati definiti nei file INCLUDE di Intuition.

Lo stesso tipo di descrizione vale ancora per i file INCLUDE dei dispositivi, rappresentati nella parte destra della Tavola 2: questi file sono necessari per definire le interazioni tra il programma e le singole routine delle librerie relative ai dispositivi. Per esempio, se il programma necessita di un'uscita audio stereo, deve aprire la libreria del dispositivo Audio (con una chiamata alla funzione OpenDevice) e inizializzare variabili di tipo struct (struttura) con i modelli di struttura definiti appositamente nei file INCLUDE del dispositivo Audio. Solo così il programma può accedere alle routine interne del dispositivo.

Le librerie dell'Amiga

Anzitutto è necessario sciogliere fin dall'inizio un dubbio che spesso s'insinua negli aspiranti programmatori dell'Amiga. Un programma apre le librerie dell'Amiga e chiama le funzioni in esse contenute soltanto quando viene mandato in esecuzione. Durante la fase di link, contrariamente a quanto molti si aspettano, vengono invece richiamate le librerie linked, di cui parleremo fra poco, e non le librerie dell'Amiga. Anche se questa considerazione può apparire banale per chi possiede più esperienza, ha ugualmente la sua importanza perché contribuisce ancora una volta a differenziare le operazioni che avvengono nella fase di compilazione e di link, dalle operazioni che avvengono quando il programma eseguibile viene mandato in esecuzione. Inoltre, la differenza tra le librerie dell'Amiga e le librerie linked non è importante solo concettualmente, ma si riflette anche nel modo in cui vengono eseguiti i programmi e sullo spazio occupato sul disco dai file eseguibili del proprio programma.

Fatta questa premessa, analizziamo meglio le libreria dell'Amiga. Se si escludono alcune librerie dei dispositivi, quasi tutte le librerie specifiche dell'Amiga sono ad accesso condiviso (shared), cioè due o più task possono aprirle e fare uso delle loro routine contemporaneamente. Alcune di queste librerie, chiamate spesso librerie del ROM Kernel, sono effettivamente contenute in ROM e quindi vengono aperte dai task senza che siano necessari accessi al disco. Ma altre sono raccolte su disco in apposite directory, e quindi aprirle significa che il sistema le deve caricare da disco nella memoria RAM. Quando per esempio si esegue un programma, e questo effettua una chiamata alla funzione OpenLibrary, l'AmigaDOS cerca nelle opportune directory su disco, o in ROM, i dati che costituiscono la libreria indicata, e quindi la apre per permettere al programma di accedere alla routine desiderata.

Anche se le librerie dell'Amiga non vengono mai aperte durante la fase di compilazione, il compilatore esegue comunque un controllo per assicurarsi che tutte le variabili, le strutture, i puntatori alle strutture e i parametri a cui si fa riferimento nel programma per accedere alle librerie siano definiti in uno dei file INCLUDE richiamati all'interno del programma: questo assicura la coerenza tra il programma e i file INCLUDE dell'Amiga e permette quindi l'uso delle routine interne alle librerie. Quindi, quando si parla di librerie dell'Amiga ci si riferisce a tutte le librerie del ROM Kernel: le sette librerie generali che iniziano con la libreria Exec e terminano con la libreria Workbench, e le dodici librerie dei dispositivi che iniziano con la libreria del dispositivo Audio e terminano con quella del dispositivo Trackdisk.

A differenza di quanto accade con le librerie linked, i codici che costituiscono le routine presenti nelle librerie dell'Amiga non entrano a far parte dei file eseguibili. Le routine di queste librerie dal punto di vista dei task sono sempre in memoria, e possono essere condivise da una pluralità di task senza conflitti. È lo stesso Exec a preoccuparsi di controllare che tutto proceda per il meglio. All'interno dei file eseguibili vengono solo indicati gli offset delle routine e l'indirizzo base della libreria che le contiene. Questa condivisione presenta l'ovvio beneficio di ridurre le dimensioni dei programmi che non sono più obbligati ad accogliere tutti i codici Assembly delle funzioni, ma solo le istruzioni che le chiamano.

Librerie di tipo linked

Una libreria linked è, come suggerisce il nome, una libreria che viene chiamata direttamente in causa durante la fase di link. È sempre contenuta in un file sul disco del compilatore, e il linker si riferisce a essa se il suo nome è stato dichiarato dal programmatore negli argomenti della linea comandi del linker. Se è stata citata, la libreria viene quindi sottoposta alla fase di link insieme a tutti i moduli già compilati del programma che si desidera realizzare. Una libreria linked contiene un insieme di funzioni che sono state preprogrammate e riunite in uno speciale file. Ogni funzione è un modulo di codice oggetto, che il linker congloba nel programma eseguibile qualora venga chiamata in qualche punto del codice sorgente. Il

vantaggio delle librerie linked offerte dal C è che una stessa libreria di funzioni precompilate può essere impiegata da una pluralità di programmi e programmatori, in modo da far risparmiare tempo ed energie. All'interno di un libreria linked, infatti, compaiono i moduli oggetto delle funzioni più comuni, più versatili e chiamate più di frequente. Un esempio è la libreria del linguaggio C standard, che viene usata da quasi tutte le varie versioni di C, e che contiene le funzioni standard previste dal C, come printf() e scanf(). Queste funzioni, come quelle di ogni altra libreria linked, sono state create utilizzando un compilatore C e un compilatore Assembly. Una volta conclusa la fase di debug, sono state poi raggruppate in un file libreria. Se il proprio programma chiama quindi le funzioni scanf o printf, si deve sottoporlo alla fase di link citando anche la libreria del linguaggio C standard.

Il processo di link, in realtà, colloca nel codice eseguibile una copia della funzione che abbiamo chiamato, cioè una copia che contiene tutte le istruzioni in linguaggio Assembly che definiscono il codice oggetto della funzione, proprio perché la funzione non è presente nelle librerie dell'Amiga. Se per esempio si scrive un programma che chiama le funzioni printf e scanf, il programma eseguibile che si ottiene dopo la fase di link include necessariamente una copia di tutte le istruzioni in linguaggio Assembly che definiscono queste due funzioni: il programma eseguibile diventerà di conseguenza più ingombrante. Questo significa che il programma eseguibile è spesso molto più grande di quello che potrebbe essere, a causa di queste funzioni delle librerie linked. In alcuni casi capita anche che le funzioni delle librerie linked siano programmate in modo tale da inserire nel programma eseguibile molti inutili codici in linguaggio Assembly (in genere opzioni che magari non verranno mai usate dal particolare programma) aumentandone ancora di più le dimensioni. Quindi, ciascun programma che esegua una chiamata a queste funzioni, si ritroverà con una copia completa di tutte le relative istruzioni Assembly, tanto quelle indispensabili quanto quelle inutili.

È ovviamente vantaggioso avere a disposizione un set preconfezionato di funzioni affidabili, ma proprio per via della loro elevata versatilità, oltre ad aumentare le dimensioni dei programmi finiscono per inserirvi codici in Assembly superflui.

Le librerie ad accesso condiviso

Tutte le librerie del ROM Kernel, salvo alcune per i dispositivi, sono ad accesso condiviso; come conseguenza, ogni struttura Library di una libreria contiene un apposito parametro che tiene sotto controllo il numero di task che accedono contemporaneamente alla relativa libreria. In genere, ogni volta che un task apre una libreria del ROM Kernel con una chiamata alla funzione OpenLibrary, il parametro lib_OpenCnt della struttura Library associata viene incrementato, per indicare che un ulteriore task sta accedendo alle funzioni di quella particolare libreria. E lo stesso parametro viene ovviamente decrementato di un'unità ogni volta che la libreria viene chiusa con una chiamata a CloseLibrary. In questo modo il sistema sa in ogni momento quanti task stanno interagendo con le librerie interne dell'Amiga.

È importante capire che se due task paralleli condividono la stessa libreria, in realtà non accedono *nello stesso preciso istante* alle sue routine: il sistema multitasking dell'Amiga infatti fa in modo che il controllo della CPU venga continuamente scambiato fra i due task. Se il primo task detiene il controllo, e chiama una funzione della libreria, potrebbe perdere il controllo prima che la funzione sia terminata, e un secondo task, quello che riceve il controllo, potrebbe chiamare un'altra o la stessa funzione della libreria. Quando il primo task perde il controllo, la funzione che aveva chiamato si interrompe, ma la libreria non viene chiusa mentre il secondo task assume il controllo della CPU. Quando il primo task riottiene nuovamente il controllo, l'esecuzione della funzione viene ripresa là dov'era stata interrotta. Questo spiega perché le funzioni dell'Amiga devono garantire la possibilità di essere sospese e in seguito riprese: si tratta della cosiddetta "multitasking code reentrancy" e viene utilizzata soprattutto sui grandi sistemi multiutente che funzionano in timesharing.

Questa capacità di uscire e rientrare, da parte di una funzione di una libreria, è una caratteristica standard del sistema software dell'Amiga, e il controllo delle precedenze è mantenuto interamente dall'Exec senza chiamare in causa il programmatore. Su questa caratteristica si può contare tranquillamente, benché sia utile essere consapevoli del modo in cui funziona e del suo significato.

Un esempio con due file

Per discutere un soggetto complesso è sempre meglio scegliere esempi semplici e schematici su cui basare la trattazione: questa sarà la filosofia su cui ci baseremo anche noi. Il primo esempio è illustrato schematicamente nella Tavola 3, che illustra un programma composto da due file. File1 contiene il modulo principale del programma – la funzione main() – e altre due funzioni che abbiamo chiamato Funzione1 e Funzione2. File2, il secondo modulo di programma, contiene tre funzioni, Funzione3, Funzione4 e Funzione5. In realtà ci troviamo di fronte a moduli scheletrici, che si limitano a rappresentare blocchi di codici solo concettualmente. Di fatto ognuna delle funzioni, compresa la funzione main(), potrebbe svolgere moltissimi compiti, ma per gli scopi che vogliamo raggiungere è superfluo complicare ulteriormente l'esempio. Quindi, le funzioni, oltre a essere molto

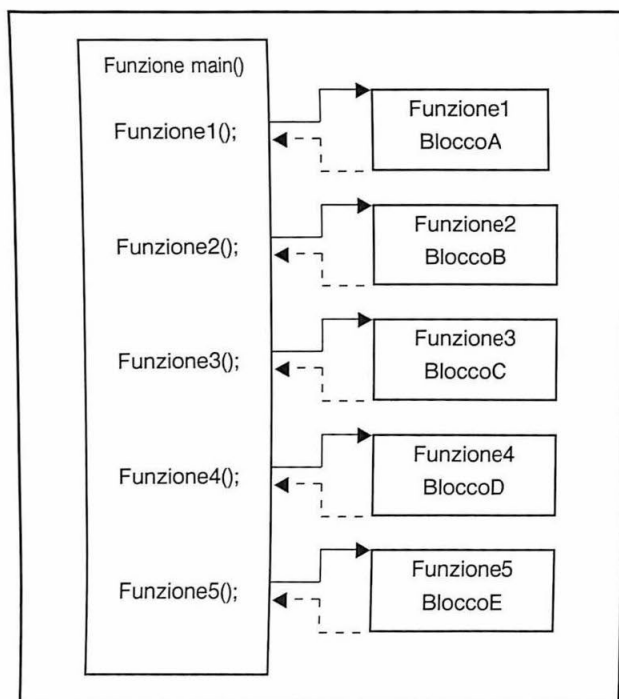



Tavola 3

semplici, non prevedono alcun argomento.

Nel nostro caso il programma è composto da due soli moduli di programma sorgente: se all'interno di questi moduli si facesse riferimento a una variabile che non è definita né in File1 né in File2, dovrebbe essere per forza definita in una delle quattro categorie di file INCLUDE precedentemente descritte. Nel nostro caso, comunque, il programma è così semplice che non contiene riferimenti a nessun file INCLUDE.

I concetti che stiamo per illustrare sono importanti non solo per imparare a costruire un programma, ma anche per capire le modalità con cui viene svolto il debug, dove vengono esaminati i dettagli del flusso di programma e la sequenza delle chiamate di ritorno. Nel nostro esempio, per visualizzare durante l'esecuzione del programma l'ordine in cui vengono chiamate ed eseguite le funzioni, sono state inserite nei due moduli sorgente alcune chiamate alla funzione standard printf() in varie posizioni. Se si desidera seguire l'esecuzione ancor più da vicino è sufficiente aggiungerne altri. Ogni funzione, quando termina, restituisce il controllo al blocco da cui è stata chiamata. Per terminare la propria esecuzione, le funzioni, data la loro semplicità, non avrebbero bisogno di alcuna istruzione particolare. Ma per completezza, nel nostro esempio le terminiamo con l'istruzione return(), che in genere serve per restituire ai codici chiamati i dati prodotti dalla funzione. L'istruzione return() permette inoltre d'interrompere una funzione in qualunque punto al suo interno, senza doverne per forza completare l'esecuzione.

Dalla figura si osserva che la funzione principale chiama ciascuna delle altre cinque funzioni in ordine progressivo, da Funzione1 a Funzione5. Le chiamate alle singole funzioni possono essere alternate a blocchi di codici che svolgono altre operazioni. In un programma realistico, uno degli scopi di queste istruzioni addizionali potrebbe essere la definizione degli appropriati valori per gli argomenti delle singole funzioni che saranno chiamate.

Il flusso di controllo procede nel modo seguente: Funzione1 viene chiamata, e il controllo del programma passa al suo punto d'ingresso (il Program Counter del 68000 viene impostato con la locazione del punto d'ingresso relativo alla prima istruzione di Funzione1). Durante la chiamata alla funzione, il sistema compie determinate operazioni che coinvolgono lo stack del task, a cui è bene dedicare qualche parola.

Lo stack del task

Lo stack del task, ovvero lo user mode stack predisposto in RAM dall'Amiga per ogni task, viene utilizzato per salvare il contenuto dei registri del Motorola 68000 appena prima che venga ceduto il controllo a una funzione. Nel nostro caso si tratta del contenuto dei registri della funzione main(), che viene salvato prima che il controllo venga ceduto a Funzione1. Oltre a questi registri della CPU, le routine di sistema in genere collocano nello user mode stack (sempre prima che il controllo venga ceduto a Funzione1) una copia dei valori relativi agli argomenti di Funzione1. Il compilatore in linguaggio C disegna automaticamente il codice in linguaggio Assembly (prodotto in realtà dall'assembler in maniera trasparente al programmatore) cosicché ciascuna funzione, quando riceve il controllo e prima di proseguire con la propria esecuzione, otterrà i parametri che si aspetta di ricevere accedendo ai valori immagazzinati nello user mode stack.

Quando la CPU termina l'esecuzione dei codici che costituiscono Funzione1, o incontra l'istruzione return(), il controllo viene restituito ai codici chiamanti appena dopo l'istruzione che ha chiamato la funzione. Per effettuare questo ritorno, il sistema accede ancora una volta alle informazioni originariamente inserite nello user mode stack per sapere a quale indirizzo di memoria ricedere il controllo. Il programmatore, comunque, non deve curarsi di queste operazioni: è compito del compilatore definire i dettagli per la traduzione del programma in linguaggio Assembly, assicurandosi che nello user mode stack siano collocati gli opportuni dati prima che il controllo venga ceduto alla funzione chiamata.

Nella sequenza in cinque fasi che abbiamo tracciato, dove ciascuna funzione contiene soltanto un'istruzione di ritorno e questa si trova esattamente alla fine di ogni funzione, l'ordine di esecuzione dei blocchi di codice è quello illustrato nella Tavola 3: BloccoA, BloccoB, BloccoC, BloccoD e finalmente BloccoE. Le istruzioni in lin-

guaggio Assembly di ogni funzione, corrispondenti alle istruzioni del sorgente in linguaggio C, vengono eseguite in modo lineare, dal punto d'ingresso fino all'istruzione di ritorno. Le linee tratteggiate nella figura rappresentano il percorso seguito quando viene mandata in esecuzione l'istruzione in linguaggio Assembly corrispondente all'istruzione return() in linguaggio C.

Un esempio più complesso

La Tavola 4 costituisce un esempio di flusso di programma leggermente più complesso, grazie a un piccolo cambiamento delle sei funzioni impiegate nel precedente esempio (suggeriamo di studiare la Tavola 4 nel corso della lettura). In questo secondo caso si vede che il controllo del programma passa da una funzione all'altra in una sequenza iniziata dalla funzione principale main(), ma da essa non completamente controllata. Infatti alcune delle funzioni chiamate da main() chiamano altre funzioni, creando una struttura di funzioni nidificate.

Una volta ancora supponiamo che ogni istruzione return() in linguaggio C si trovi al termine di ciascuna funzione. Rispetto all'esempio precedente, la sola differenza è che tre funzioni (Funzione2, Funzione3 e Funzione4) contengono un'altra istruzione in linguaggio C che chiama la funzione successiva: Funzione2 chiama Funzione3, Funzione3 chiama Funzione4, e Funzione4 chiama infine Funzione5. Vedremo ora che queste nuove istruzioni sconvolgono completamente il flusso del programma rispetto all'esempio precedente.

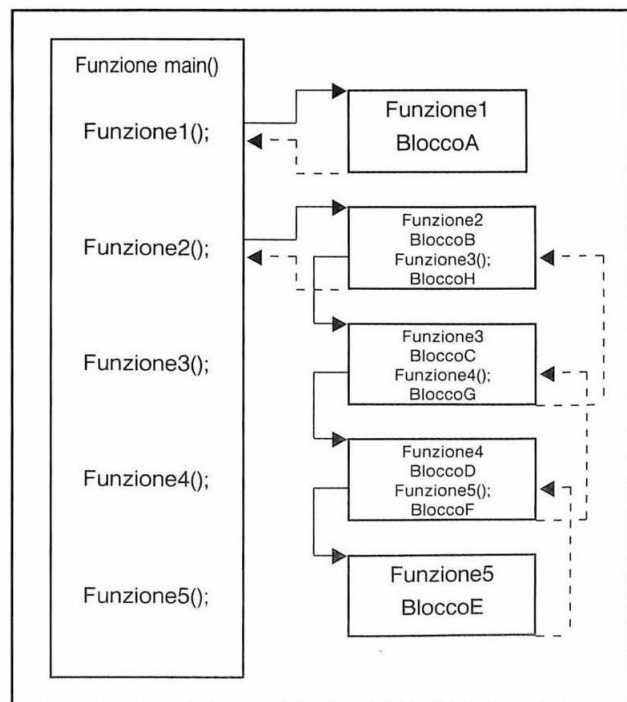
La sequenza delle operazioni si può così riassumere. La funzione main() chiama Funzione1,

Tavola 4

questa completa la sua esecuzione e restituisce il controllo alla funzione principale. Questo primo gradino è esattamente identico a quello del primo esempio. Poi la funzione main() chiama Funzione2, ma questa viene eseguita solo fino al momento in cui manda in esecuzione Funzione3: il controllo a quel punto passa infatti a Funzione3. A sua volta, Funzione3 viene eseguita soltanto fino al punto in cui chiama Funzione4, la quale riceve il controllo e viene eseguita soltanto fino al punto in cui chiama Funzione5, cedendole il controllo. Si noti che nessuna di queste funzioni, eccetto Funzione5, incontra ed esegue l'istruzione di ritorno in tutto il percorso finora analizzato. In questo momento dell'esecuzione ci aspettiamo che nello user mode stack siano presenti gli indirizzi di ritorno che individuano le istruzioni successive alle chiamate in Funzione4, Funzione3, Funzione2, e main().

Funzione5 viene eseguita fino all'ultima istruzione, quella di ritorno. Ma questa volta il controllo non torna alla funzione principale, come accadeva nell'esempio precedente: le linee tratteggiate mostrano in dettaglio il percorso seguito per il ritorno. La complicazione è questa: i cinque gradini illustrati precedentemente, quando Funzione2 chiamava Funzione3 e così via, vengono ripercorsi a ritroso. Le linee tratteggiate della Tavola 4 mostrano che il percorso di ritorno da Funzione5 alla funzione principale (che prima avveniva direttamente) ora si complica attraversando quattro gradini. Questa sequenza di chiamate a ritroso può essere meglio capita partendo dall'ultima funzione della sequenza e procedendo all'indietro. Quando Funzione5 arriva all'ultima istruzione (quella di ritorno), restituisce il controllo a Funzione4 al punto immediatamente successivo a quello in cui era avvenuta la chiamata di Funzione5, e così via fino a tornare passo dopo passo a main().

La chiamata della funzione main() a Funzione2 ha quindi come risultato complessivo l'esecuzione di tutti i codici di Funzione2, Funzione3, Funzione4 e Funzione5, ma l'ordine in cui ora vengono eseguiti i vari blocchi di codice è il seguente: BloccoA, BloccoB, BloccoC, BloccoD, BloccoE, BloccoF, BloccoG, BloccoH, come appunto si vede nella Tavola 4. Si può confrontare questo nuovo ordine con quello mostrato nella Tavola 3, osservando però che i blocchi di codice sono etichettati in modo leggermente diverso nei due esempi. Tutti questi cambiamenti nella sequenza d'attività del programma dipendono soltanto dalla posizione reciproca delle chiamate alle funzioni e delle istruzioni di ritorno (e, nel nostro caso, anche dall'aggiunta delle quattro istruzioni di chiamata alle altre funzioni che prima non erano presenti).

A questo punto main() prosegue con la chiamata a Funzione3 e quindi a Funzione4 e Funzione5, le quali produrranno ancora chiamate nidificate. Infine, la funzione principale esegue una chiamata alla funzione exit() e il programma termina.

L'esecuzione della funzione exit() è necessaria solo se si desidera restituire un dato al termine del programma, altrimenti si può evitare d'inserirla al termine di main().

L'esecuzione pratica

Per realizzare in pratica i due esempi, analizziamo i passi successivi che si devono compiere. Assumiamo che il lettore possieda il pacchetto *Aztex Manx C*, il compilatore al quale faremo sempre riferimento in questo corso. Nelle prossime puntate scenderemo maggiormente nei dettagli, ma per il momento ci interessa arrivare rapidamente a osservare i risultati dei due esempi.

Dal momento che entrambi gli esempi sono composti da due moduli sorgente, i passi da seguire valgono per entrambi. Anzitutto occorre digitare i codici sorgente in due file di testo differenti che chiamiamo rispettivamente File1.c e File2.c. Per svolgere questa operazione, è possibile servirsi di un qualunque editor. Se si è incerti nella scelta, si può sempre usare ED, l'editor disponibile da CLI su tutti i dischi sistema. Quando si è digitato il primo file sorgente, prima di procedere con il secondo si può aumentare l'efficienza della programmazione sfruttando le capacità multitasking dell'Amiga. Si impartisca da CLI la seguente linea comando:

```
RUN CC File1.c
```

In questo modo la compilazione del primo modulo sorgente, quello principale, viene avviata come processo in background, e nel frattempo possiamo iniziare a digitare il file sorgente relativo al secondo modulo di programma. Al termine digitiamo ancora la seguente linea comando:

```
CC File2.c
```

Se non si sono verificati errori, sul disco sono ora presenti due file oggetto di nome File1.o e File2.o. Non rimane che sottoporli alla fase di link per ottenere il file eseguibile. Si digiti la seguente linea comando:

```
LN -o Programma File1.o File2.o -lc
```

Ora su disco il linker ha creato il file eseguibile, che abbiamo chiamato Programma, riunendo i due moduli oggetto e risolvendo tutti i riferimenti incrociati, con l'ausilio ovviamente delle librerie linked. Se ora digitiamo:

```
Programma
```

il programma viene mandato in esecuzione, e sullo schermo appariranno tutti i messaggi che abbiamo inserito nei codici sorgente tramite le funzioni printf().

Non aggiungiamo altro a questo stadio. Questi pochi passi conducono alla compilazione dei due programmi d'esempio, ma ci siamo arrivati senza spiegare cosa effettivamente si stava facendo. Nel corso delle prossime puntate, ognuno di questi passi sarà ampiamenente discusso.

Listato 1 (relativo alla Tavola 3)

Definizione dei contenuti di File1.c:

```
main()
{
 Funzione1();
 Funzione2();
 Funzione3();
 Funzione4();
 Funzione5();
 exit();
}

Funzione1()
{
 printf("Ora e' in esecuzione Funzione1, BloccoA\n");
 /* Qui si trova il blocco di codice BloccoA */
 return();
}

Funzione2()
{
 printf("Ora e' in esecuzione Funzione2, BloccoB\n");
 /* Qui si trova il blocco di codice BloccoB */
 return();
}
```

Definizione dei contenuti di File2.c:

```
Funzione3()
{
 printf("Ora e' in esecuzione Funzione3, BloccoC\n");
 /* Qui si trova il blocco di codice BloccoC */
 return();
}

Funzione4()
{
 printf("Ora e' in esecuzione Funzione4, BloccoD\n");
 /* Qui si trova il blocco di codice BloccoD */
 return();
}

Funzione5()
{
 printf("Ora e' in esecuzione Funzione5, BloccoE\n");
 /* Qui si trova il blocco di codice BloccoE */
 return();
}
```

Listato 2 (relativo alla Tavola 4)

Definizione dei contenuti di File1.c:

```
main ()
{
 Funzione1();
 Funzione2();
 Funzione3();
 Funzione4();
 Funzione5();
 exit();
}

Funzione1()
{
 printf("Ora e' in esecuzione Funzione1, BloccoA\n");
 /* Qui si trova il blocco di codice BloccoA */
```

```
  return();
}

Funzione2()
{
  printf("Ora e' in esecuzione Funzione2, BloccoB\n");
  /* Qui si trova il blocco di codice BloccoB */
  Funzione3();
  printf("Ora e' in esecuzione Funzione2, BloccoH\n");
  /* Qui si trova il blocco di codice BloccoH */
  return();
}
```

Definizione dei contenuti di File2.c:

```
Funzione3()
{
  printf("Ora e' in esecuzione Funzione3, BloccoC\n");
  /* Qui si trova il blocco di codice BloccoC */
  Funzione4();
  printf("Ora e' in esecuzione Funzione3, BloccoG\n");
  /* Qui si trova il blocco di codice BloccoG */
  return();
}

Funzione4()
{
  printf("Ora e' in esecuzione Funzione4, BloccoD\n");
  /* Qui si trova il blocco di codice BloccoD */
  Funzione5();
  printf("Ora e' in esecuzione Funzione4, BloccoF\n");
  /* Qui si trova il blocco di codice BloccoF */
  return();
}

Funzione5()
{
  printf("Ora e' in esecuzione Funzione5, BloccoE\n");
  /* Qui si trova il blocco di codice BloccoE */
  return();
}
```

La prossima puntata

Nella seconda puntata del corso discuteremo da cima a fondo l'organizzazione ottimale di un efficiente sistema di programmazione in linguaggio C per l'Amiga. Discuteremo la scelta di un buon editor per i file sorgente, l'uso di un programma di controllo sintattico per verificare la sintassi dei file sorgente, l'impiego del compilatore, dell'assembler, del linker e del *Source Level Debugger* della Manx. La completa analisi degli strumenti di lavoro di cui è necessario disporre è infatti un passo fondamentale per programmare in C con l'Amiga. ■

COME PROGRAMMARE I GADGET STRINGA DI INTUITION

Iniziamo un'analisi dei gadget dell'Amiga ideata per correggere gli errori del manuale di Intuition e completarlo. In questo articolo, con l'aiuto di un breve programma in C, esaminiamo i gadget stringa, i loro parametri e i relativi bug

di Harriet Maybeck Tolly

Il manuale di Intuition (l'*Amiga Intuition Reference Manual*) parla dei gadget come degli elementi fondamentali di Intuition. I gadget forniscono un'interfaccia tramite la quale l'utente inserisce dati e comunica con le applicazioni. Tra i gadget più comuni possiamo citare i gadget stringa (utilizzati ad esempio per digitare i nomi dei file), i gadget di sovrapposizione delle finestre e i gadget proporzionali (detti anche barre di scroll) che permettono per esempio lo scroll all'interno di una finestra dei nomi dei file presenti su un disco.

Quasi tutti i programmi ricorrono ai gadget di Intuition per scambiare informazioni con l'utente, ma ne esistono alcuni che creano gadget personalizzati dotati di particolari comportamenti non previsti da Intuition. A volte, infatti, i programmatori hanno bisogno di gadget che si comportino in modi molto particolari, e ovviamente Intuition non sempre può aiutarli. Ecco perché qualche volta i gadget hanno reazioni diverse da quelle che ci aspetteremmo. Per esempio, per una di queste "imitazioni" la combinazione Amiga-Destro/X potrebbe non avere il solito effetto di cancellare il contenuto del gadget stringa attivo in quel momento.

I gadget disponibili sono di tre tipi:

1) gadget stringa, che accettano stringhe di testo alfanumerico (esiste un sottotipo di gadget stringa, il gadget numerico, che accetta soltanto numeri interi)

2) gadget booleani, detti anche "toggle", che possono assumere soltanto due stati

3) gadget proporzionali, o barre di scroll.

Il manuale di Intuition è indispensabile per chi desidera dedicarsi alla programmazione dei gadget, dal momento che descrive dettagliatamente tutte le strutture che devono essere definite. Ma benché indispensabile, a volte non è sufficiente.

Questo articolo non è un semplice riassunto del manuale: anzi, le informazioni contenute nel manuale non vengono neanche citate, se non quando è strettamente necessario. La nostra intenzione è invece quella di coprire aree poco chiare, o che contengono qualche errore.

Definizione dei gadget stringa

Intuition, nel suo complesso, è una struttura davvero notevole, grazie soprattutto ai gadget, tramite i quali il programmatore può realizzare con facilità interfacce utente particolarmente efficaci. Per ora limitiamo il nostro campo d'interesse ai gadget stringa, riservandoci di prendere in esame anche gli altri nei prossimi articoli.

Chiunque abbia dovuto rispondere a un requester che chiedeva il nome di un file sa che cos'è un gadget stringa. Quando si digita il nome di un file, infatti, non si fa altro che inserire dati in un gadget stringa. Il disegno che rappresenta questo tipo di gadget può essere complesso come quello del gadget *English* del programma *SpeechToy* con il suo triplo bordo, o può essere del tutto inesistente.

Un gadget il cui parametro GadgetType è impostato a STRGADGET (cioè un gadget stringa) deve avere il parametro SpecialInfo della struttura Gadget che punta a una struttura di tipo StringInfo. La struttura StringInfo è così definita nel file INCLUDE intuition.h:

```
struct StringInfo {
  UBYTE    *Buffer;
  UBYTE    *Undobuffer;
```

```
  SHORT    BufferPos;
  SHORT    MaxChars;
  SHORT    DispPos;
  SHORT    UndoPos;
  SHORT    NumChars;
  SHORT    DispCount;
  SHORT    CLeft,CTop;
  struct   Layer *LayerPtr;
  LONG     LongInt;
  struct   KeyMap *AltKeyMap;
};
```

Alcune voci presenti in questa struttura hanno bisogno di qualche spiegazione.

Buffer è un puntatore a un'area di memoria contenente la stringa che il gadget visualizza inizialmente, cioè il testo proposto per default che l'utente può considerare valido, alterare oppure cancellare interamente. Questa stringa, chiamata stringa iniziale, dev'essere predisposta dal task ed è molto importante che finisca con il carattere NULL, cioè che l'ultimo byte sia uno zero, altrimenti il gadget continua a visualizzare tutti i dati che seguono in memoria. Se si inserisce la propria stringa inizializzandola nel modo seguente:

```
char mystring[] = {"stringa di prova"};
```

il compilatore C provvederà automaticamente a concluderla con uno zero. Se invece si prevede di dover assegnare il valore dinamicamente, è necessario ricordarsi d'inserire come ultimo byte il valore NULL. Se si desidera che la stringa iniziale sia vuota, come spesso accade, bisogna semplicemente impostare il primo carattere a "\0".

Il parametro Buffer individua anche l'area di memoria nella quale il gadget memorizza la stringa digitata dall'utente, e di conseguenza nella sua definizione in C bisogna dichiarare uno spazio sufficiente ad accogliere la quantità di caratteri indicata dal parametro MaxChars della struttura StringInfo. Per esempio, se Buffer punta a mystring, che ha lunghezza 16, e MaxChars vale 100, gli 84 caratteri che seguono il buffer mystring in memoria verranno sovrascritti dal testo che l'utente digita oltre i primi 16 caratteri. Viceversa, MaxChars non dev'essere di lunghezza inferiore a quella della stringa inizialmente puntata da Buffer: se Buffer punta a mystring e MaxChars vale solo 5, viene ugualmente visualizzata l'intera stringa, senza badare alla grandezza del rettangolo che dovrebbe contenerla. Non ci sono invece problemi se la larghezza dell'area di schermo occupata dal gadget non riesce a contenere l'intero testo presente nel buffer: Intuition provvede infatti a visualizzare solo quanto è possibile e a gestire lo scroll orizzontale del testo.

UndoBuffer punta a un secondo buffer che contiene la stessa stringa a terminazione nulla presente nel buffer individuato dal parametro Buffer. L'aggiornamento di questo secondo buffer rispetto al primo viene effettuato ogni volta che l'utente seleziona il gadget con il mouse, o preme Return quando il gadget è selezionato. Il suo compito è mantenere una copia della stringa contenuta nel buffer prima dell'ultima variazione, in modo da realizzare la funzione undo; questa funzione, attivabile premendo la combinazione di tasti Amiga-Destro/Q, consente di far riapparire nel gadget l'ultimo testo inserito (se Return non è stato premuto neanche una volta, si tratta della stringa iniziale). Se invece si preferisce che il proprio programma non abbia alcuna capacità undo, basta impostare a zero il puntatore UndoBuffer. È un modo per risparmiare memoria, anche se così non si rende certo un buon servizio all'utente.

Il buffer puntato dal parametro UndoBuffer dovrebbe sempre essere della stessa capienza del buffer individuato dal parametro Buffer. Inoltre, dal momento che Intuition permette che sia attivo un solo gadget stringa alla volta, uno stesso buffer di ripristino (o buffer undo) può essere utilizzato per diversi gadget stringa (in questo caso si deve fare in modo che la sua capienza sia uguale a quella del gadget stringa più esteso). Sebbene questo metodo si possa considerare sicuro e permetta di risparmiare memoria, potrebbe rivelarsi scomodo. Supponiamo infatti che due gadget stringa impieghino lo stesso buffer undo: se l'utente seleziona uno dei due, vi digita qualcosa e poi seleziona l'altro, il contenuto del buffer undo relativo al primo gadget viene automaticamente perso quando l'utente seleziona il secondo; se successivamente l'utente riattiva il primo gadget, Intuition ripristina il contenuto del buffer undo con l'ultima versione del testo. In questo modo, per il primo gadget, e implicitamente anche per il secondo, l'originale stringa per la funzione undo viene sicuramente persa. Se invece ognuno dei due gadget dispone di un proprio buffer undo, il programmatore può realizzare una gestione manuale della funzione undo, preservando il testo iniziale di entrambi.

MaxChars è il numero massimo di caratteri che l'utente può digitare nella stringa più lo zero conclusivo. Abbiamo già parlato precedentemente dei valori che questo parametro può assumere. Se per un nuovo gadget stringa il parametro Buffer punta a uno spazio di memoria di dimensioni diverse da quelle del precedente, anche MaxChars dev'essere aggiornato di conseguenza.

I tre parametri della struttura StringInfo che stiamo ora per descrivere sono del tipo "creati dal programmatore, mantenuti da Intuition". Questo significa che una volta ceduto a Intuition il controllo dei gadget, non si devono più cambiarne i valori. Se proprio non si potesse farne a meno, è necessario ricorrere alla funzione RemoveGadget (o eventualmente RemoveGList) prima del cambiamento, e alla funzione AddGadget (o AddGList) dopo il cambiamento, in modo da restituire momentaneamente il controllo del gadget al task e poter operare in tutta sicurezza (si veda a questo proposito il listato che accompagna l'articolo).

BufferPos è la posizione iniziale del cursore nel

buffer. Il manuale afferma che il valore di questo parametro viene mantenuto da Intuition, ma l'affermazione è vera soltanto finché non si varia il parametro Buffer perché punti a un'altra area di memoria. Quando si altera Buffer, si deve infatti stare bene attenti alla presenza di un vecchio BufferPos non idoneo. Per sicurezza, conviene nella maggior parte dei casi reimpostarlo a zero ogni volta che si alloca un nuovo buffer. Il task che ha definito il gadget può sempre accedere a questo parametro per conoscere l'attuale posizione del cursore.

DispPos è la posizione iniziale nel buffer del primo carattere da visualizzare. Anche di questo si prende cura Intuition, ma soltanto fino a quando il parametro Buffer non viene alterato. Nel dubbio, conviene reimpostarlo sempre a zero per ogni nuovo gadget. Il task che ha definito il gadget può sempre accedere a questo parametro per sapere qual è il primo carattere a sinistra, all'interno del gadget stringa.

NumChars è il numero di caratteri contenuti nel buffer. È lo stesso Intuition a calcolarlo, quando il gadget del programmatore viene visualizzato. Intuition ne mantiene in memoria il valore finché il puntatore Buffer non viene alterato.

DispCount è il numero di caratteri che possono essere interamente visualizzati nel rettangolo che delimita l'area di schermo occupata dal gadget. È un valore che viene calcolato automaticamente da Intuition, e garantisce la corretta visualizzazione del testo anche quando si definisce per il gadget un rettangolo la cui larghezza non è un multiplo esatto della larghezza del carattere. Questo parametro non dovrebbe essere mai variato dal task. Se qualcuno è tanto dispettoso da cambiare la larghezza del rettangolo dopo che il gadget è stato visualizzato, Intuition dovrebbe avere la gentilezza di ricalcolare il valore del relativo parametro, ma non consiglierei a nessuno di provare.

La struttura Gadget per i gadget stringa

Ora diamo un'occhiata ai passi da seguire per inizializzare la struttura Gadget e creare un gadget stringa. Ricordiamo che qualunque sia il gadget che si vuole creare, bisogna sempre inizializzare una struttura di tipo Gadget, definita nel file INCLUDE intuition.h. Le osservazioni che seguono riguardano soltanto gli elementi della struttura Gadget che a nostro avviso possono non essere del tutto chiari.

- La costante GADGHIGHBITS per il parametro Flags: questa costante è una maschera che Intuition paragona al valore di Flags impostato dal task per stabilire quale modo highlight è stato richiesto (cioè in che modo Intuition deve segnalare all'utente l'avvenuta selezione del gadget: complemento di colore del rettangolo, ulteriore cornice, immagine o bordo alternativo, nessuna segnalazione). Si ricordi a questo proposito che una maschera è un valore binario che viene utilizzato in un'operazione logica AND per azzerare i bit di un dato che non sono d'interesse. Per sua natura, quindi, questa costante non dev'essere usata in un'operazione OR con il parametro Flags, cioè per impostare l'uno o l'altro bit (flag). Può invece essere utile a un task che deve controllare il valore dei flag che indicano il modo highlight selezionato. In questo caso, il tipo d'istruzione da impiegare è la seguente:

```
highlight_attuale = GADGHIGHBITS & Gadget.Flags;
```

nella quale si suppone che le variabili siano state definite come segue:

```
USHORT highlight_attuale;
struct Gadget Gadget;
```

- Il flag SCRGADGET di GadgetType: sorpresa! I gadget di schermo non esistono, questo flag non serve a nulla.
- MutualExclude: altra sorpresa! La funzione di questo parametro della struttura Gadget non è mai stata implementata in Intuition. Il manuale aggiornato della versione 1.2 illustra comunque un sistema per simulare la mutua esclusione, ma si può applicare soltanto ai gadget booleani – per ora – e quindi ne parleremo nel prossimo articolo di questa serie.
- Il flag SELECTED del parametro Flags: se per un gadget booleano di tipo toggle è impostato a 1, il gadget viene visualizzato nel modo highlight (evidenziato). La funzione di questo flag è quindi visualizzare un gadget booleano in modo cha appaia già selezionato. Questo flag è privo di significato per i gadget stringa e quelli proporzionali. Se si desidera che un gadget stringa appaia già selezionato, in modo da non rendere necessaria la sua selezione con il mouse, si deve chiamare la funzione ActivateGadget.
- Il flag GRELWIDTH del parametro Flags: se è impostato a 1 indica a Intuition che il valore memorizzato nel parametro Width non rappresenta la larghezza dell'area occupata dal gadget, ma l'offset negativo che dev'essere sommato alla coordinata x del lato destro dell'elemento di schermo contenente il gadget (finestra o requester) per ottenere il limite estremo destro del gadget (se l'offset è positivo il gadget oltrepassa il limite destro dell'elemento di schermo in cui è contenuto). In pratica, se il flag GRELWIDTH è azzerato, il parametro Width costituisce la larghezza assoluta del gadget, mentre se è impostato a 1 individua il limite destro del gadget prescindendo dalla posizione del suo limite sinistro. Quindi, valori negativi del parametro Width hanno senso solo se il flag GRELWIDTH è già stato impostato. Lo stesso vale per il flag GRELHEIGHT e il parametro Height.

Alcuni elementi della struttura Gadget richiedono valori estremamente precisi. Per fare un esempio citerò un caso non documentato da nessun manuale, ma del tutto vero, almeno secondo le esperienze di chi scrive. Supponiamo che il nostro task visualizzi un gadget nello stato

disabilitato (gadget fantasma) e che al momento giusto lo abiliti (l'utente può selezionarlo con il mouse solo quando è abilitato). Supponiamo inoltre che sia stato impostato il seguente valore:

```
Gadget.Width = 162
```

e che sia stata eseguita l'istruzione OR GADGDISABLED con il parametro Gadget.Flags. A questo punto il task apre la finestra, e il gadget viene visualizzato, ma non è attivo (gadget fantasma); sino a questo momento è tutto sotto controllo. Ora il task decide che è il momento di abilitare il gadget, rendendo superfluo che lo faccia l'utente. Deve rimuovere il gadget, azzerare il flag GADGDISABLED eseguendo l'istruzione AND ~ GADGDISABLED con Gadget.Flags, e infine ripristinare nuovamente il gadget sullo schermo. E iniziano i problemi. Si nota infatti che all'estrema destra del rettangolo ci sono strani rimasugli che ho affettuosamente battezzato "Fuzzies", piccole briciole di fantasma che non sono state eliminate nel momento in cui il nostro gadget è ricomparso sullo schermo.

Secondo la mia esperienza, la presenza di Fuzzies dipende dalla larghezza dell'area occupata dal gadget, e può essere evitata facendo in modo che il parametro Gadget.Width contenga un multiplo esatto della larghezza della fonte che si sta utilizzando. Tornando all'esempio precedente, con la fonte Topaz 80 la larghezza del carattere è 8 pixel, e quindi per evitare i Fuzzies dobbiamo dichiarare una larghezza di 160 (20 caratteri per 8 pixel), e non di 162 com'è stato indicato. Per verificarlo, è sufficiente alterare in 197 il quarto parametro (192) della quinta struttura StrGadget e ricompilare il programma.

Un'altra caratteristica curiosa dei gadget stringa è la sindrome del raddoppiamento dell'ultimo carattere. Provate a eseguire questa sequenza di comandi: mandate in esecuzione il tool *IconEd* dal *Workbench*. Selezionate Load dal menu, premete il pulsante del mouse sul gadget stringa, digitate abbastanza caratteri da riempire il buffer (quando raggiungete la massima capienza del buffer lo schermo lampeggia). Ora premete Shift/Tasto-Cursore-Sinistro: quest'operazione porta il cursore all'estrema sinistra del gadget e la stringa al primo carattere, quindi ora state osservandone l'inizio (prendete nota del carattere all'estrema destra del buffer). Ora premete Shift/Tasto-Cursore-Destro, portandovi al margine destro della stringa: l'estremità destra della stringa dovrebbe trovarsi all'immediata sinistra del cursore e questo dovrebbe trovarsi su una casella vuota. Sorpresa! Quel carattere che vi avevo suggerito di notare se ne sta tranquillamente accovacciato sotto il cursore.

Purtroppo non ho trovato nessun modo per eliminare questo bug. Ne ho fatto cenno qui soltanto perché siate preparati psicologicamente, quando apparirà sui vostri gadget: non è colpa vostra, ma di Intuition.

Uso dei gadget stringa da parte del programma e dell'utente

1. ActivateGadget()

```
Success = ActivateGadget (Gadget, Window, Requester)
```

Questa funzione è stata una delle novità più interessanti della versione 1.2 del software sistema: basta chiamarla e il gadget stringa diventa attivo, proprio come se l'utente l'avesse selezionato con il mouse. Utilizzandola opportunamente non si costringe più l'utente a premere il pulsante del mouse sull'area del gadget prima di potervi digitare un testo.

I programmatori devono tenere conto che una chiamata ad ActivateGadget non ha successo se è attiva una qualunque finestra diversa da quella contenente il gadget. Ecco quali sono le condizioni che devono essere rispettate perché la chiamata abbia successo:

A) la finestra contenente il gadget dev'essere attiva

B) nessun gadget dev'essere in uso (cioè bisogna che l'utente non stia variando la grandezza della finestra proprio in quel momento, per esempio)

C) se il gadget stringa si trova in un requester, il requester dev'essere attivo

D) il pulsante destro del mouse non dev'essere premuto (cioè bisogna che l'utente non stia visualizzando il menu proprio in quel momento).

Basta che una sola di queste condizioni non sia rispettata e la chiamata non ha successo. Se la continuazione del programma dipende dalla riuscita di quest'operazione, ci si deve ricordare di controllare il valore di ritorno prodotto dalla funzione.

2. Premendo la combinazione di tasti che attiva i menu (Amiga-Destro/Alt-Destro) oppure il tasto destro del mouse, il gadget eventualmente attivo viene automaticamente disattivato.

Il listato

Il listato che accompagna questo articolo è un esempio di programma in C che apre una finestra contenente cinque gadget stringa. La voce Scroll Su del menu fa scorrere verso l'alto dieci testi all'interno dei cinque gadget, mentre la voce Scroll Giù li fa scorrere verso il basso. Ogni volta che si preme il Return all'interno di un gadget, il programma segnala in quale gadget è stato premuto. Si noti che per la larghezza (Width) del primo gadget, indicata nella struttura StrGadget è stato impostato il flag GRELWIDTH del parametro Flags; l'ultimo gadget stringa è un gadget fantasma: la voce Fantasma del menu lo abilita e lo disabilita; tutte le larghezze dei gadget sono multipli di otto: se volete vedere un po' di Fuzzies basta aggiungere 2 alla larghezza dell'ultimo gadget e selezionare la voce Fantasma.

Lasciamo ora la parola al programma. Per il mese prossimo, l'appuntamento è con i gadget booleani.

```
MIClasse     = IntuiMessage->Class;
MICodice     = IntuiMessage->Code;
MIIndirizzo = IntuiMessage->IAddress;
ReplyMsg(IntuiMessage);
/* rileva la ragione del messaggio */
switch (MIClasse)
{
   case MENUPICK: /* tasto menu del mouse */
      while (MICodice != MENUNULL)
      {
         numvoce = ITEMNUM(MICodice);
         switch (numvoce)
         {
         case 2:  /* voce Fantasma */
            /* prima di alterare si rimuove il  */
            /* gadget dal controllo di Intuition*/
            gad_pos = RemoveGList(Finestra,
                        &StrGadget[4], 1L);
/* Esegue un OR esclusivo per invertire lo stato del flag
GADGDISABLED. Non usa le funzioni OnGadget e OffGadget (che
produrrebbero lo stesso effetto) in quanto chiamano
RefreshGadgets. Questa routine ha effetto su TUTTI i gadget
a partire nella lista da quello indicato, causandone lo
sfavillio sullo schermo */
            StrGadget[4].Flags ^= GADGDISABLED;
            /* restituisce il gadget a Intuition */
            real_pos = AddGList(Finestra,
              &StrGadget[4], (LONG)gad_pos,
              1L, (LONG)NULL);
            /* rivisualizza il gadget */
            RefreshGList(&StrGadget[4], Finestra,
              (LONG)NULL, 1L);
            break;
         case 1: /* scroll in alto dei 10 testi */
            if (k == 9) k = 0; else k++;
            /* prima di alterare si rimuove il */
            /* gadget dal controllo di Intuition */
            gad_pos = RemoveGList(Finestra,
                        &StrGadget[4], 5L);
            for (i = 0; i<5; i++)
            {  StringInfo[i].Buffer =
                  (UBYTE *)buffer_gad[(i+k)%10];
               StringInfo[i].BufferPos = 0;
               StringInfo[i].NumChars  = 0; }
            /* restituisce il gadget a Intuition */
            real_pos = AddGList(Finestra,
              &StrGadget[4], (LONG)gad_pos,
              5L, (LONG)NULL);
            /* rivisualizza il gadget */
            RefreshGList(&StrGadget[4], Finestra,
              (LONG)NULL, 5L);
            break;
         case 0: /* scroll in basso dei 9 testi */
            if (k == 0) k = 9; else k--;
            /* prima di alterare si rimuove il */
            /* gadget dal controllo di Intuition */
            gad_pos = RemoveGList(Finestra,
                        &StrGadget[4], 5L);
            for (i = 0; i<5; i++)
            {  StringInfo[i].Buffer =
                  (UBYTE *)buffer_gad[(i+k)%10];
               StringInfo[i].BufferPos = 0;
               StringInfo[i].NumChars  = 0; }
            /* restituisce il gadget a Intuition */
            real_pos = AddGList(Finestra,
              &StrGadget[4], (LONG)gad_pos,
              5L, (LONG)NULL);
            /* rivisualizza il gadget */
            RefreshGList(&StrGadget[4], Finestra,
              (LONG)NULL, 5L);
            break;
         } /* switch, fine analisi dei menu */
         /* controlla se e' ancora stato */
         /* premuto il tasto del mouse */
         MICodice = (ItemAddress(&gadget_menu,
                     MICodice))->NextSelect;
      } /* while */
      break;
   case CLOSEWINDOW: /* chiusura della finestra */
      /* restituisce qualunque messaggio in coda */
      while (IntuiMessage = (struct IntuiMessage *)
            GetMsg(Finestra->UserPort))
         ReplyMsg(IntuiMessage);
      MiElimino();
      exit(TRUE);
      break;
   case GADGETUP: /* pressione RET in un gadget */
      igad = (struct Gadget *)MIIndirizzo;
      ID   = igad->GadgetID;
      printf("RETURN premuto nel gadget %d\n",ID);
      break;
   default:
      puts("Messaggio da Intuition non previsto");
      break;
   } /* switch */
      }     /* while  */
    }       /* if     */
  }         /* for    */
}           /* main   */

MiElimino()
{  if (Finestra)       CloseWindow(Finestra);
   if (GfxBase)        CloseLibrary(GfxBase);
   if (IntuitionBase) CloseLibrary(IntuitionBase); }
```

LA STRUTTURA EXECBASE DELL'AMIGA

Nell'Amiga esiste un direttore esecutivo che controlla l'intero sistema: la libreria Exec. Per poter intervenire sugli aspetti fondamentali della macchina è necessario conoscere la sua struttura ExecBase, la più importante fonte d'informazioni sul sistema

di Eugene P. Mortimore

Ogni azienda ha al suo vertice un uomo chiave che deve tenere costantemente sott'occhio tutte le attività in corso. Quest'uomo deve anche riesaminare di tanto in tanto la struttura dei quadri aziendali, e mantenersi aggiornato sulla situazione dei singoli dipartimenti e dei loro sottodipartimenti, e su ogni variazione organizzativa. È il direttore esecutivo dell'azienda: la sua vigilanza costante e competente è indispensabile per consentire alla ditta di muoversi con efficienza. Il suo lavoro non si limita alla supervisione delle direttive impartite dai superiori, ma prevede anche la possibilità d'impartire ordini diretti per migliorare con rapidità l'intero processo produttivo.

Se l'azienda è di grandi dimensioni il suo lavoro aumenta di conseguenza, e per svolgerlo il direttore esecutivo ha bisogno di altre persone: una segretaria, un vicedirettore e alcuni capireparto.

La situazione che abbiamo descritto può essere considerata un'ottima metafora di quanto avviene nell'Amiga. Anche il nostro computer ha un "uomo" chiave, e quest'uomo viene chiamato appunto direttore esecutivo. Si tratta del sistema Exec o, più sinteticamente, dell'Exec, una libreria di routine che costituisce una parte importante del sistema operativo dell'Amiga. Come il nostro immaginario direttore esecutivo, l'Exec è sia il comandante in capo, sia l'onnipresente sorvegliante di tutti gli eventi in corso che, di momento in momento, cambiano l'organizzazione interna dell'Amiga. Con l'unica differenza che la sua sorveglianza non è giornaliera, ma è regolata su tempi che hanno come ordine di grandezza la frazione di microsecondo.

Se avviene un cambiamento nell'hardware del computer, se per esempio il cuore stesso del sistema viene sostituito (supponiamo che il microprocessore cambi dal 68000 al 68020), l'Exec dev'esserne informato, e deve fare in modo che tutti i task siano immediatamente in grado di ricevere la notizia e di sfruttarla a proprio vantaggio. Andando oltre, l'Exec dev'essere in grado di effettuare cambiamenti che abbiano effetto sull'intero "schema organizzativo" del computer. Dev'essere addirittura in grado di mettere in atto cambiamenti tanto radicali che il computer a volte può sembrare, dal punto di vista del mondo esterno, una macchina completamente diversa. L'Exec ideale è onnisciente, onnipresente e onnipotente.

Lo schema organizzativo dell'Amiga

Proprio come avviene per l'organizzazione del personale di un'azienda, l'Exec dell'Amiga ha bisogno di una "mappa" che consenta di tenere costantemente sotto sorveglianza le attività e le continue variazioni di stato del sistema. A questo scopo, nell'Amiga esiste la struttura ExecBase. Questa struttura contiene tutte le informazioni di cui l'Exec ha bisogno per conoscere lo stato del sistema, e contemporaneamente fornisce anche i parametri su cui l'Exec può intervenire per cambiare il modo di operare della macchina nella sua globalità, nell'ambito di un singolo programma, o addirittura di un singolo task. I programmatori che desiderano concentrare le proprie energie sull'Amiga, devono conoscere la struttura ExecBase per sapere in che modo vengono gestite le informazioni presenti nel sistema. L'Exec deve mantenere un costante controllo diretto sulle complesse interazioni fra tutti i task che hanno accesso alla CPU e le altre risorse del sistema legate

all'hardware. Se così non fosse, ciascun task potrebbe spodestare le routine dell'Exec dettando legge per conto suo senza nessun coordinamento, e il sistema scivolerebbe rapidamente nel caos. Nella macchina regnerebbe l'anarchia.

Il sistema Exec deve garantire, per esempio, che se due task richiedono simultaneamente la stessa risorsa di sistema, questa risorsa non venga offerta ad ambedue nello stesso momento; deve assicurarsi che vengano stabiliti dei turni, in accordo con le leggi scolpite nelle sue routine. L'Exec ha sempre l'ultima parola.

I comportamenti dei task

Quando sono in esecuzione, i task possono svolgere molte operazioni. Possono richiedere e liberare memoria di sistema dinamicamente; disabilitare (e in seguito riabilitare) gli interrupt hardware, interrompendo momentaneamente le interazioni della macchina con il mondo esterno; arrestare momentaneamente la gestione multitasking; aggiungere al sistema nuove librerie generali e librerie per i dispositivi di I/O, e aprirle di loro iniziativa; mandare in esecuzione un'ampia varietà di funzioni del ROM Kernel; infine, creare message port per lo scambio di messaggi e segnali con gli altri task dello stesso programma, o con i task di altri programmi (in questo caso le message port dei due programmi devono essere opportunamente strutturate, e quindi fra loro compatibili, in modo da permettere un corretto scambio di dati).

Oltre a queste operazioni che coinvolgono direttamente l'Exec, nel sistema Amiga è importante anche lo "stato" di ogni task. Un task può essere pronto per l'esecuzione, per esempio; in questo caso il momento in cui riceve il controllo della CPU dipende dalle routine dell'Exec che costituiscono l'algoritmo di ripartizione della CPU fra tutti i task eseguibili (il cosiddetto scheduler). Un task in questo stato può rimanere pigramente in attesa anche per lungo tempo, se la sua priorità d'esecuzione è molto bassa. Oppure, un task può trovarsi nello stato di attesa di un segnale software, che ne faccia riprendere l'esecuzione quando si verifica una particolare condizione nel sistema. In questo caso si dice che il task non è eseguibile, e quindi non viene preso in considerazione dall'algoritmo di ripartizione della CPU.

Alcuni task tentano di compiere operazioni molto più sofisticate, come appropriarsi del controllo dell'intera macchina, in modo da poterla guidare al di fuori dei consueti modi operativi. Per esempio, alcuni task potrebbero tentare d'intervenire nelle normali procedure di attivazione del sistema, sostituendo le procedure di attivazione della macchina predefinite in ROM con nuove funzioni allocate in RAM. Altri potrebbero alterare queste procedure in modo che il consueto boot da disco venga eseguito per esempio da un RAM disk, o da un chip ROM situato in un dispositivo hardware esterno.

Spingendosi ancora più in là, alcuni task potrebbero generare particolari dati appositamente studiati per rendere più sofisticata e precisa la procedura di debug di un programma; altri potrebbero cambiare i dati che definiscono l'alert di Guru Meditation, estromettendo il confuso messaggio predefinito in ROM. Altri ancora potrebbero effettuare operazioni rivolte alla personalizzazione della macchina che il tipico programmatore medio non avrà mai bisogno di tentare.

Al contrario di altri sistemi, l'Amiga offre una notevole flessibilità e libertà di programmazione, grazie alla quale i programmatori più abili possono personalizzare la macchina direttamente. Come vedremo, parte di questa flessibilità nasce direttamente dalle informazioni contenute nei parametri della struttura ExecBase. Questa struttura rappresenta nell'Amiga il principale meccanismo di controllo globale del sistema. È appositamente progettata per mantenere sempre disponibile tutta una serie di vitali parametri che caratterizzano lo stato del sistema in ogni momento. Le routine della libreria Exec accedono continuamente a queste informazioni, sia in lettura sia in scrittura, per mantenere l'ordine nel mutevolissimo ambiente multitasking dell'Amiga.

Cosa devono sapere sistema e programmatore?

Il sistema multitasking Exec ha bisogno di sapere quali sono le componenti hardware di cui è costituita la macchina. Per esempio di quale CPU (68000, 68010, 68020, o 68030) e di quale coprocessore matematico (68881 o 68882) è dotato il sistema, dove si trovano in memoria tutte le strutture del sistema Exec e le relative informazioni di supporto, quali strutture sono state allocate e inizializzate nella RAM dai programmi che hanno ricevuto il controllo durante la fase di attivazione del *Kickstart* o dopo la sua conclusione. L'Exec mantiene traccia di tutti questi tipi di strutture tramite le liste di sistema. Ha bisogno inoltre di sapere in ogni istante quale task detiene il controllo della CPU.

Oltre a questo, l'Exec ha bisogno di sapere per quale Paese è stata progettata la macchina, cioè se si tratta di un Paese che prevede una rete elettrica a 50 Hz e il sistema PAL europeo, o una rete elettrica a 60 Hz e il sistema NTSC americano. E ancora deve sapere se la normale procedura di attivazione è stata modificata rispetto a quella predefinita in ROM, se la memoria del sistema è stata opportunamente configurata, e altro ancora. Vediamo invece di quali conoscenze dovrebbe disporre un programmatore esperto.

Un programmatore può aver bisogno di rilevare quanta memoria è disponibile nella macchina (sia nell'area della memoria chip, ovvero i primi 512K, sia in quella della memoria fast, o estesa), d'intervenire sulla normale procedura di attivazione per configurare la macchina secondo i suoi scopi, di

saper sistemare le cose in modo che il boot possa essere eseguito da RAM disk... e di moltissime altre risposte a interrogativi che vanno oltre la normale struttura dei task.

I primi argomenti

In questo articolo intendiamo fare alcune considerazioni generali sulla programmazione dell'intero sistema. Daremo un'occhiata d'insieme alle routine della libreria Exec e cercheremo di rispondere a due fondamentali domande: «Come fa un task definito dal programmatore a interagire con le routine dell'Exec?» e «Come fa un task a sfruttare le informazioni presenti in alcune delle strutture previste dall'Exec, citate nei file INCLUDE, per interagire con le routine di sistema?». Discuteremo ciascun parametro della struttura ExecBase e vedremo come questi parametri forniscono utili informazioni alle routine interne di sistema o ai task definiti dal programmatore.

Va precisato che vengono discussi soltanto i primi 16 parametri della struttura ExecBase (da LibNode a IntVects). In questo gruppo sono compresi la maggior parte dei cosiddetti "parametri statici": si tratta di variabili che in generale non cambiano dopo che il sistema è stato attivato. I rimanenti parametri della struttura verranno discussi nel prossimo articolo. Si ricordi infine che una discussione simile può essere applicata a ciascuna delle "strutture di mantenimento" che controllano le principali librerie di sistema: la struttura GfxBase per la libreria Graphics, la struttura IntuitionBase per la libreria Intuition e così via. Alcune di queste strutture verranno descritte nei prossimi articoli.

L'importanza del codice sorgente Commodore

Il codice sorgente utilizzato dalla Commodore per creare le routine interne del sistema Amiga non è disponibile, e non è quindi consultabile dal generico programmatore: per la Commodore è una proprietà preziosa, quindi non è di pubblico dominio e probabilmente non lo sarà mai. Almeno, questa è la posizione attuale della Commodore. Per di più, le routine interne del sistema cambiano di continuo, via via che appaiono nuove versioni del software sistema: la compatibilità è sempre assicurata, ma i dettagli del loro modo di operare restano celati a tutti i programmatori che non lavorano per la Commodore. A noi queste routine appaiono come una scatola nera e per scoprire che cosa c'è dentro possiamo soltanto cercare di capire come funzionano le più importanti strutture del sistema. ExecBase è una delle più interessanti.

La segretezza del codice sorgente rende difficile accertare con sicurezza come sono definiti alcuni dei parametri della struttura e come variano nel corso delle varie fasi in cui vengono eseguite le routine dei task e le routine interne di sistema. Per la maggior parte dei programmatori, d'altra parte, una così approfondita conoscenza del sistema non è necessaria; è sufficiente che conoscano il significato dei parametri di cui intendono servirsi. In ogni caso, se si vuole conoscere il sistema meglio del programmatore medio – cioè capire in ogni dettaglio l'architettura dell'Amiga – è inutile cercare di mettersi nelle mani della Commodore. L'approccio che seguiremo nel nostro articolo sarà esaminare i parametri della struttura ExecBase, la più importante, e dedurne il modo di operare del sistema. Non arriveremo a dare risposta a tutte le possibili domande, ma metteremo sul tappeto una notevole quantità di materiale su cui riflettere.

Condizioni globali del sistema

Quando si tenta di capire il significato dei vari parametri presenti nella struttura ExecBase, è importante ricordare che questa struttura rappresenta le condizioni globali della macchina. Con questa espressione vogliamo dire che i parametri della struttura ExecBase caratterizzano da ogni punto di vista le condizioni operative del sistema, sia nel corso della sua evoluzione, sia nel momento in cui un task detiene il controllo della CPU, sia quando il mondo esterno interviene nelle operazioni della macchina attraverso gli interrupt hardware.

La Tavola 1 offre un quadro generale della struttura ExecBase. Le informazioni documentate dai suoi parametri possono essere raggruppate in alcune categorie principali.

– Informazioni che si riferiscono direttamente al task in esecuzione; il puntatore ThisTask individua la struttura Task che rappresenta il task in esecuzione, mentre i parametri TaskTrapCode, TaskExitCode, TaskExceptCode, TaskSigAlloc e TaskTrapAlloc rappresentano particolari informazioni sul task in esecuzione.

– I nomi delle più importanti liste del sistema: MemList, ResourceList, DeviceList, IntrList, LibList, PortList, TaskReady, TaskWait e SemaphoreList. Queste liste di sistema sono particolarmente importanti perché legano tra loro la maggior parte delle strutture di dati presenti nel sistema, separandole per argomenti. Scorrendo le liste di sistema, i task possono facilmente individuare e consultare i dati di qualsiasi struttura.

– Informazioni sullo stack di sistema: i parametri SysStkUpper e SysStkLower. Esiste un unico stack di sistema, il supervisor mode stack del 68000, che dev'essere abbastanza grande da riuscire a contenere tutte le informazioni da salvare durante l'esecuzione dei codici del sistema operativo, e delle routine di interrupt hardware e di trap. Oltre a questo stack globale, ciascun task possiede il proprio stack personale, detto anche user mode stack, che viene definito nella struttura Task associata al task.

– Informazioni relative a dati allocati in RAM che normalmente il sistema si aspetta di trovare in ROM. Stiamo parlando dei codici e dei dati creati

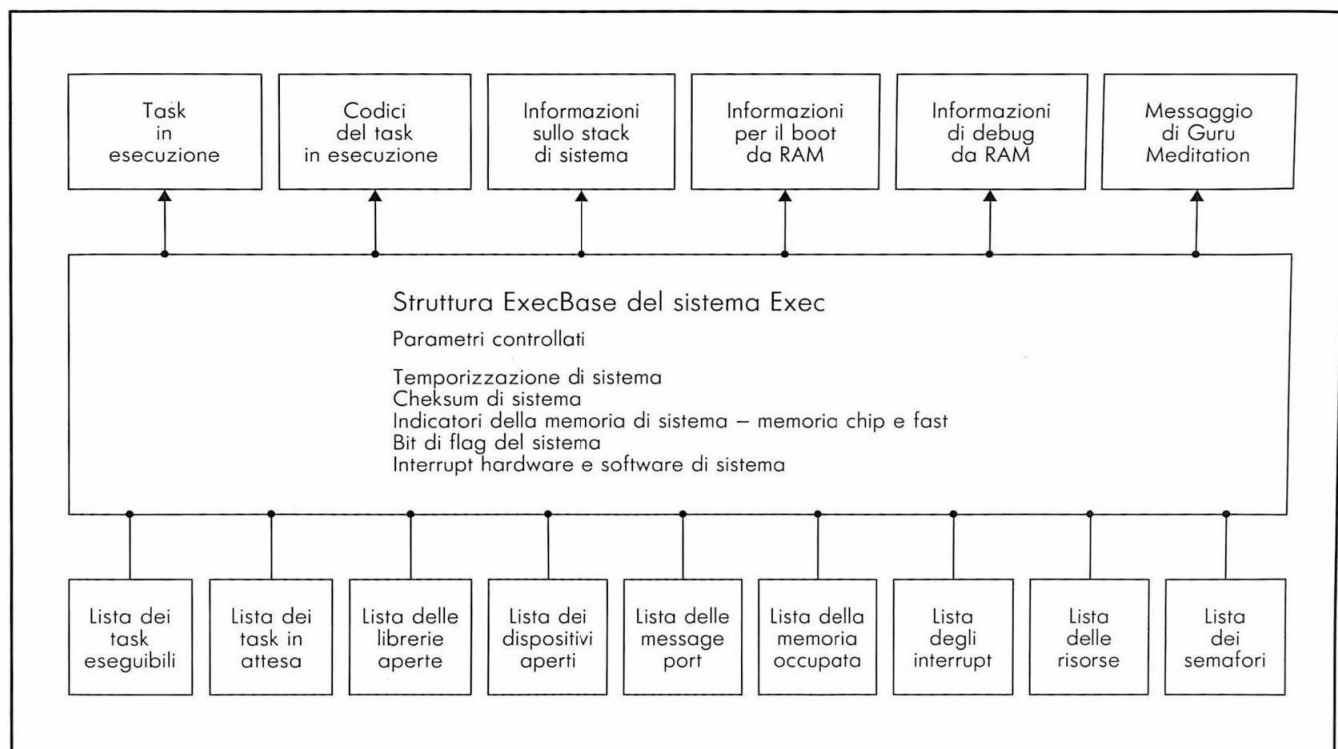



Tavola 1

dal programmatore in RAM per scavalcare o espandere le procedure di sistema predefinite in ROM. Cambiando per esempio il contenuto dei parametri KickMemPtr, KickTagPtr e KickCheckSum, è possibile imporre che il boot venga effettuato attraverso dati e routine presenti in RAM, eludendo così la normale procedura predefinita in ROM.

Nello stesso modo si può agire sul parametro DebugData in modo che il ROMWack Debugger acceda ai nostri dati in RAM anziché a quelli predefiniti nella ROM. Agendo su questi parametri della struttura ExecBase, il programmatore può quindi personalizzare completamente la macchina, escludendo o espandendo le procedure standard predefinite in ROM.

– Informazioni relative alle condizioni di abilitazione o disabilitazione degli interrupt hardware del sistema: il parametro IDNestCnt. Per ottenere maggiori informazioni su questo parametro, si consulti la definizione della struttura Task nel testo *Programmare l'Amiga Vol. I* (in questo libro sono illustrati nei dettagli molti degli argomenti qui affrontati).

– Informazioni sullo stato della gestione multitasking del sistema, cioè lo scambio di controllo fra i task il cosiddetto task switching (si consulti a questo proposito l'articolo "La gestione multitasking, i tipi di codice e gli interrupt", apparso sul numero 5/88 di *Commodore Gazette*): il parametro TDNestCnt.

– Informazioni relative alla temporizzazione degli eventi basata sulla scansione del video, e alla frequenza dell'alimentazione di rete: i parametri VBlankFrequency e PowerSupplyFrequency. Questi parametri indicano implicitamente anche la nazionalità, americana o europea, di ciascuna macchina.

– Informazioni relative alla partizione di tempo della CPU assegnata a ogni task in una serie di task dotati della stessa priorità, e alla frequenza dello scambio di controllo fra i task: i parametri Elapsed e Quantum.

– Informazioni sui moduli residenti predefiniti nel sistema: il parametro ResModules. Questi moduli di programma residenti permettono di costruire e conservare nel sistema insiemi predefiniti di utili routine, in modo che non debbano essere ogni volta compilate e sottoposte al linker insieme ai programmi. Questo permette notevoli economie sulle dimensioni dei file eseguibili ed è un ulteriore esempio dell'aiuto che il sistema fornisce ai programmatori. Ci soffermeremo ancora sui moduli residenti nel prossimo articolo.

– Un insieme d'informazioni relative alla continua evoluzione dell'intero sistema: i parametri SoftVer, LowMemChkSum, ChkBase, ColdCapture, CoolCapture, WarmCapture, MaxLocMem, DebugEntry, MaxExtMem, LastAlert, ChkSum, IntVects, AttnFlags e AttnResched. Questi parametri saranno discussi dettagliatamente in questo articolo e nel prossimo.

Come avviene l'ingresso di un task nel sistema

La Tavola 2 illustra come un task entra a far parte del sistema, come viene inserito nella lista di sistema dei task eseguibili, e come ottiene infine –

eventualmente dopo una certa attesa – una partizione di tempo (time slice) della CPU. In generale, i task vengono inseriti nella lista di sistema dei task eseguibili dalla funzione AddTask della libreria Exec. I tre argomenti di questa funzione sono rispettivamente un puntatore alla struttura Task che caratterizza il task, il nome in linguaggio C o Assembly del punto d'ingresso del task (cioè l'indirizzo della prima istruzione del task), e infine il nome in linguaggio C o Assembly del punto d'ingresso della routine di uscita dal task (cioè l'indirizzo della prima istruzione della routine).

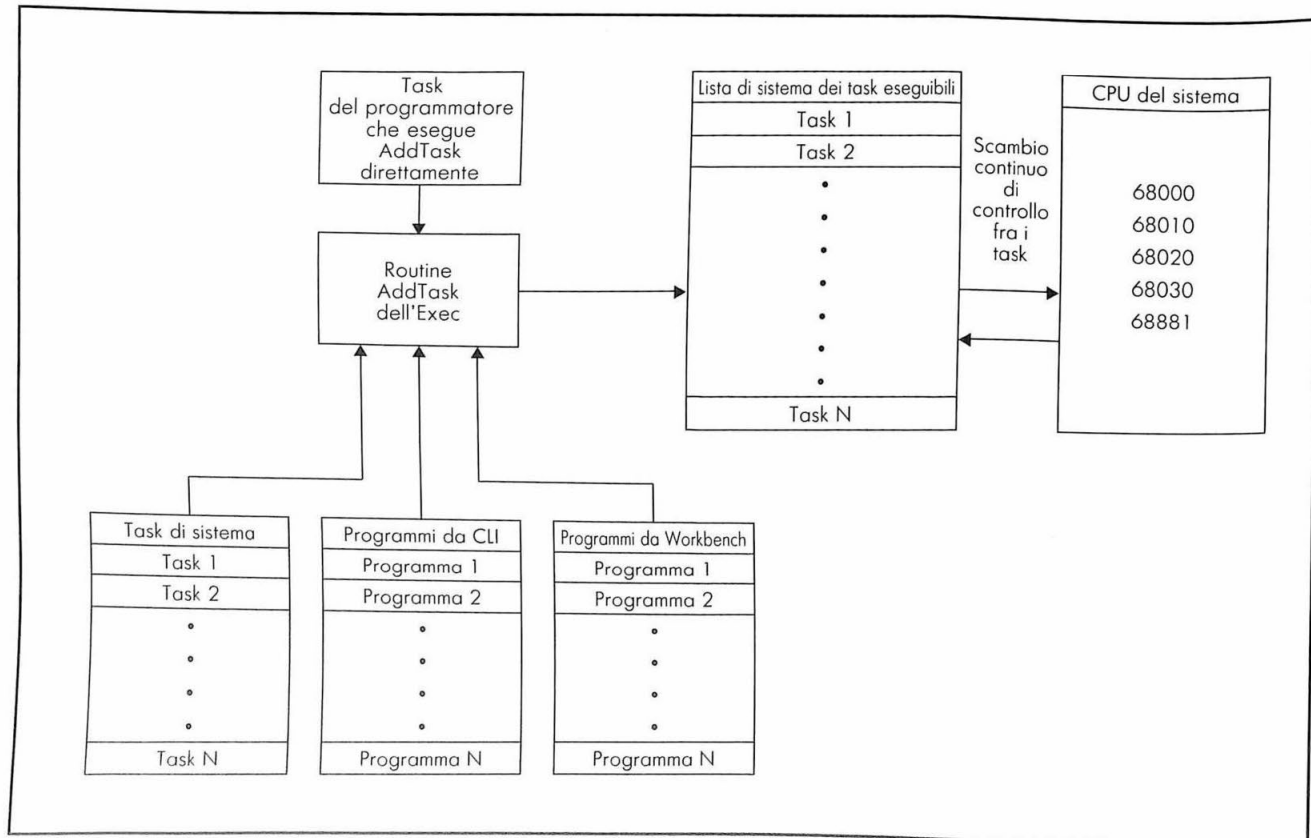
I task che appaiono nella lista di sistema dei task eseguibili sono classificabili in quattro categorie.

Per primi vengono i task "di sistema", ovvero quelli creati dal sistema stesso al momento dell'attivazione e durante la successiva gestione multitasking. Includono i cosiddetti task "file system", i task "input device", i task "trackdisk device", i task "CON" e – se è attivo un RAM disk – i cosiddetti task "RAM". In genere questi task vengono creati automaticamente dal sistema e sono esterni al controllo diretto del programmatore o dell'utente.

I task di secondo tipo sono i programmi mandati in esecuzione da CLI (Command Line Interface, interfaccia linea comando). Si tratta dei programmi che l'utente manda in esecuzione richiamandoli dalla linea comando di una finestra CLI. L'utente può aprire tutte le finestre CLI che vuole, impartendo ripetutamente il comando NEWCLI dell'AmigaDOS. Ogni finestra corrisponde a un CLI in esecuzione, indipendente da tutti gli altri. Ancora una volta è il sistema stesso che assegna a questi task la relativa priorità e li aggiunge automaticamente alla lista dei task eseguibili ricorrendo alla funzione AddTask o a un suo equivalente (l'utente può sempre cambiare la priorità assegnata a un task utilizzando il comando SetTaskPri dell'AmigaDOS).

Al terzo tipo di task appartengono quelli mandati in esecuzione agendo sullo schermo del *Workbench*. In genere per avviare questi programmi bisogna premere due volte il pulsante del mouse sulla relativa icona (si può trattare dell'icona che individua il programma, oppure dell'icona project-file che ne individua un file di dati). Anche per questi tipi di task la priorità viene assegnata automaticamente dal sistema, che provvede anche ad aggiungere il task alla lista di sistema dei task eseguibili ricorrendo alla funzione AddTask o a un suo equivalente.

Nel quarto tipo di task rientrano quelli definiti all'interno dei programmi creati dal programmatore, che devono essere aggiunti al sistema con esplicite chiamate alla funzione AddTask effettuate all'interno delle routine dei programmi. In questo caso la priorità del singolo task dipende dal valore che il programma inserisce nel parametro ln_Pri della sotto-struttura Node appartenente alla struttura Task.

Tavola 2

La struttura ExecBase

In questo articolo cerchiamo di mettere in luce gli aspetti generali del sistema: considerando tutte le routine interne della libreria Exec come una raccolta di routine non direttamente visibili, ci chiediamo quali restrizioni ciascun task del programmatore incontri nell'interfacciarsi con esse attraverso i parametri della struttura ExecBase.

In questo senso, i file INCLUDE realizzati dalla Commodore costituiscono una sorgente d'informazioni davvero utile. Nella programmazione dell'Amiga la loro consultazione è fondamentale, in quanto oltre a svolgere un ruolo determinante nella compilazione dei programmi, i file INCLUDE rappresentano anche la documentazione ufficiale di tutte le strutture riconosciute dall'Amiga, dei relativi parametri e dei bit di flag.

Per quanto riguarda la struttura ExecBase (definita nel file INCLUDE execbase.h), per ora a noi interessa soprattutto capire come le routine dei task definiti dal programmatore possano accedere in scrittura e in lettura ai suoi parametri, e a quali limitazioni devono sottostare per eseguire questi accessi. La definizione della struttura ExecBase, che prevede 50 parametri, è la seguente:

```
struct ExecBase {
  struct  Library LibNode;
  UWORD   SoftVer;
  WORD    LowMemChkSum;
  ULONG   ChkBase;
  APTR    ColdCapture;
  APTR    CoolCapture;
  APTR    WarmCapture;
  APTR    SysStkUpper;
  APTR    SysStkLower;
  ULONG   MaxLocMem;
  APTR    DebugEntry;
  APTR    DebugData;
  APTR    AlertData;
  APTR    MaxExtMem;
  UWORD   ChkSum;
  struct  IntVector IntVects[16];
  struct  Task *ThisTask;
  ULONG   IdleCount;
  ULONG   DispCount;
  UWORD   Quantum;
  UWORD   Elapsed;
  UWORD   SysFlags;
  BYTE    IDNestCnt;
  BYTE    TDNestCnt;
  UWORD   AttnFlags;
  UWORD   AttnResched;
  APTR    ResModules;
  APTR    TaskTrapCode;
  APTR    TaskExceptCode;
  APTR    TaskExitCode;
  ULONG   TaskSigAlloc;
  UWORD   TaskTrapAlloc;
  struct  List MemList;
  struct  List ResourceList;
  struct  List DeviceList;
  struct  List IntrList;
  struct  List LibList;
  struct  List PortList;
  struct  List TaskReady;
  struct  List TaskWait;
  struct  SoftIntList SoftInts[5];
  LONG    LastAlert[4];
  UBYTE   VBlankFrequency;
  UBYTE   PowerSupplyFrequency;
  struct  List SemaphoreList;
  APTR    KickMemPtr;
  APTR    KickTagPtr;
  APTR    KickCheckSum;
  UBYTE   ExecBaseReserved[10];
  UBYTE   ExecBaseNewReserved[20];
};
```

Le variabili dinamiche e statiche della struttura ExecBase

In ambiente Amiga i parametri di struttura possono essere classificati in diversi modi. I parametri che variano nel tempo vengono denominati "parametri dinamici" o "variabili dinamiche" del sistema. In generale questi parametri vengono inizializzati ai loro valori di default dalle routine della libreria Exec durante la fase di attivazione della macchina, ma in seguito continuano a variare nel corso delle interazioni tra le routine di sistema, i task di sistema e i task definiti dal programmatore.

Oltre ai parametri dinamici esiste un'altra classe di parametri: i cosiddetti "parametri statici". Sono parametri di struttura che vengono generalmente inizializzati dal sistema durante la fase di attivazione della macchina, e mantengono il loro valore per un periodo di tempo molto lungo, talvolta addirittura per tutto il periodo in cui la macchina non viene spenta o sottoposta al reset. Spesso sono proprio costanti caratteristiche della macchina, e rappresentano grandezze invariabili. Si tratta di una classe di parametri nettamente diversi da quelli dinamici, che invece cambiano ogni volta che il controllo passa da un task all'altro e quindi, in pratica, 16 (Quantum) volte al secondo. Alcuni dei parametri statici servono per effettuare un controllo sull'integrità del sistema; in questo gruppo, in particolare, vi sono alcuni parametri di checksum che vengono utilizzati dal sistema per verificare che le informazioni contenute in RAM siano intatte in ogni istante in cui la macchina è in funzione, e in particolare che siano intatte durante la procedura di attivazione.

Il parametro SoftVer della struttura ExecBase, per esempio, rappresenta il numero di versione del software contenuto in ROM, ovvero il numero di versione del *Kickstart*. Dopo la prima fase di attivazione, le routine di sistema esaminano il software sistema installato in ROM e ne deducono il numero di versione (questo numero è inserito in ROM nel corso del processo di fabbricazione della macchina). Le routine di sistema, una volta che ne sono entrate in possesso, lo memorizzano nel parametro SoftVer della struttura ExecBase, in modo che la versione attuale del software sistema sia sempre consultabile. Questo valore dovrebbe rimanere costante senza eccezioni: né le routine di sistema, né quelle dei task definiti dal programmatore dovrebbero alterarlo.

Nella maggior parte dei casi, i parametri che possono essere considerati statici (almeno dal momento in cui la procedura di attivazione si è completata) sono i seguenti: SoftVer, Low-

MemChkSum, ChkSum, ColdCapture, CoolCapture, WarmCapture, SysStkUpper, SysStkLower, MaxLocMem, MaxExtMem, KickMemPtr, KickTagPtr, KickCheckSum, VBlankFrequency e PowerSupplyFrequency.

Esaminiamo ora uno per uno i parametri statici della struttura ExecBase.

Come viene gestita la libreria Exec? LibNode è il nome della sotto-struttura Library tramite la quale vengono gestite le routine della libreria Exec. Si noti che il parametro LibNode, così come viene definito nella struttura ExecBase, è il nome della sotto-struttura, non il puntatore alla sotto-struttura. L'intestazione di una libreria, così come viene intesa dal sistema Exec, in generale è costituita da tre aree di memoria contigue: nell'ordine, una tavola di vettori di salto (jump vector) che individuano l'indirizzo assoluto di ogni routine presente nella libreria, una struttura di tipo Library utilizzata per definire e gestire le funzioni e i dati della libreria, e infine alcuni dati caratteristici della libreria che di solito sono impiegati solo dalle sue routine interne.

La tavola dei vettori di salto contiene una serie d'istruzioni JMP a locazioni assolute della memoria; quindi ogni vettore di salto è composto da sei byte, due per contenere il codice dell'istruzione, e quattro per l'indirizzo assoluto della corrispondente routine. Per individuare l'indirizzo di un vettore di salto, occorre sommare all'indirizzo della struttura Library, detto indirizzo base della libreria, il relativo offset (un offset è un numero relativo da sommare a un indirizzo assoluto per ottenerne uno nuovo; si dice che l'offset costituisce un indirizzo relativo a una locazione assoluta). Dal momento che nella memoria la tavola dei vettori di salto di una libreria precede sempre la relativa struttura Library, tutti gli offset delle routine della libreria sono sempre negativi. L'impiego degli offset per individuare i vettori di salto delle routine presenti nella libreria permette di mantenere un unico indirizzo assoluto per tutta la libreria.

Dopo la tavola dei vettori di salto segue la struttura Library associata alla libreria, il cui primo elemento individua anche l'indirizzo base dell'intera libreria. Infine, seguono i dati di cui le routine della libreria hanno bisogno per funzionare correttamente. Per individuare questi dati si utilizzano offset positivi rispetto all'indirizzo base.

Nel caso della libreria Exec, la tavola dei vettori di salto inizia, rispetto all'indirizzo base, con il vettore della funzione privata Supervisor (con il termine "privata" si intende dire che in condizioni normali la funzione non dovrebbe essere usata dai programmatori, ma solo dal sistema), individuabile tramite l'offset minore (–30), e termina con il vettore della funzione "pubblica" CopyMemQuick (con il termine "pubblica" si intende dire che la funzione è a disposizione dei programmatori), individuabile con l'offset maggiore (–630).

Dopo il primo vettore di salto della tavola (quello della funzione Supervisor), seguono altri sei vettori che individuano altrettante funzioni private: ExitIntr, Schedule, Reschedule, Switch, Dispatch ed Exception. I vettori che puntano a funzioni pubbliche cominciano con InitCode e continuano fino a CopyMemQuick. Scendendo ai dettagli, i vettori di salto della libreria Exec sono ordinati come segue.

Sette vettori iniziali per le già citate funzioni private Commodore. I vettori destinati alla più generale classe di funzioni pubbliche, che iniziano con InitCode e finiscono con Debug. I vettori per le funzioni relative agli interrupt, che iniziano con Disable e finiscono con Cause. I vettori per le funzioni di gestione della memoria, che iniziano con Allocate e finiscono con FreeEntry. I vettori per le funzioni di gestione delle liste, che iniziano con Insert e finiscono con FindName. I vettori per le funzioni di gestione dei task, che iniziano con AddTask e finiscono con FreeTrap. I vettori per le funzioni di gestione delle message port, che iniziano con AddPort e finiscono con FindPort. I vettori per le funzioni di gestione delle librerie, che iniziano con AddLibrary e finiscono con SumLibrary. I vettori per le funzioni di gestione dei dispositivi, che iniziano con AddDevice e finiscono con AbortIO. I vettori per le funzioni di gestione delle risorse, che iniziano con AddResource e finiscono con OpenResource. I vettori per le funzioni aggiunte nella penultima versione del sistema Exec, che iniziano con RawIOInit e finiscono con OpenLibrary. I vettori per le funzioni di supporto ai semafori della nuova versione 1.2, che iniziano con InitSemaphore e finiscono con RemSemaphore. Infine, i vettori per le funzioni di supporto alla memoria e alla ROM della versione 1.2, che iniziano con SumKickData e finiscono con CopyMemQuick.

Le funzioni contenute nella libreria Exec sono allocate in ROM nello spazio di memoria che inizia dalla locazione 0x00FC0000 e termina alla locazione 0x00FC3507.

Come si può notare, queste funzioni occupano meno di 14K di memoria ROM (per la precisione 13576 byte, nella versione 1.2.1). Grazie alle sintetiche e sofisticate tecniche di programmazione utilizzate per queste funzioni, si può parlare di un pacchetto di funzioni estremamente compatto e altamente ottimizzato.

Qual è la versione attuale del software sistema? SoftVer è il parametro che contiene il numero di versione del *Kickstart* residente nella memoria ROM della macchina. Questo numero viene scritto in ROM direttamente durante la fabbricazione; viene letto e memorizzato in SoftVer dalle routine di sistema durante la fase di attivazione. In genere SoftVer viene usato solo dalle routine interne del sistema Exec per identificare la versione dei codici in ROM. Può capitare che anche un task definito dal programmatore vada a leggere questo parametro, per mettere il programma in condizione di

decidere quali funzioni del ROM Kernel può chiamare: le prime versioni del ROM Kernel, infatti, non includevano tutte le funzioni che sono ora disponibili.

Come fa la macchina a eseguire un controllo interno? LowMemCheckSum rappresenta il valore di checksum che il sistema rileva prendendo in considerazione un particolare insieme di valori presenti nella parte iniziale della memoria RAM, dove risiede la tavola dei vettori da quattro byte assegnati alle exception del Motorola 68000 (si tenga presente che la Motorola attribuisce al termine exception un significato diverso rispetto a quello sottinteso dal manuale del ROM Kernel; si consulti il precedente articolo di questa serie sul numero 5/88 di *Commodore Gazette*). Il checksum è eseguito dalle routine dell'Exec e la descrizione dell'algoritmo impiegato non è di pubblico dominio. Per questo motivo non è possibile stabilire con precisione quali vettori di trap sono coinvolti nell'operazione.

Ogni volta che viene eseguita la procedura di riattivazione del sistema, viene effettuato questo checksum, e il risultato viene confrontato con il valore "atteso" per verificare che i vettori di trap nella RAM contengano i valori appropriati: il successo di questo controllo dimostra che la memoria RAM di sistema non è stata danneggiata in alcun modo dall'esecuzione errata di un programma nel corso della procedura di attivazione.

ChkBase è il complemento del cosiddetto "puntatore alla base del sistema". Viene usato solo internamente e non è d'interesse per i programmatori.

I successivi quattro parametri sono variabili di tipo APTR (Absolute memory pointer). Le variabili APTR sono usate principalmente per individuare i codici e i relativi dati associati a specifici tipi di task o routine di sistema. Per esempio, i parametri tc_TrapCode e tc_TrapData della struttura Task sono variabili APTR, e individuano rispettivamente i codici e l'area dati della routine di trap software relativa a un task.

Come avviene l'attivazione della macchina? I tre parametri che stiamo per analizzare sono altrettanti vettori ai quali il sistema accede durante le varie fasi della procedura di attivazione per stabilire se deve seguire la procedura standard o una differente procedura allocata in RAM da un task. Se in uno di essi è presente il valore NULL (0x00000000), il sistema prosegue con la procedura standard; in caso contrario ne deduce che la procedura predefinita dev'essere ignorata e bisogna cedere il controllo alla particolare routine che un task ha previsto in RAM. Quindi, questi vettori contengono normalmente il valore zero, e i task possono alterarli se desiderano modificare la procedura di attivazione predefinita in una o più delle sue fasi.

Quando durante l'avvio del sistema viene fatto ricorso alle routine interne che costituiscono la procedura di attivazione standard, il sistema attraversa tre fasi distinte: la prima fase è associata con il vettore di cold-capture (il parametro ColdCapture), la seconda fase con il vettore di cool-capture (il parametro CoolCapture), e la terza fase con il vettore di warm-capture (il parametro WarmCapture). Ognuna di queste fasi standard può essere sostituita da procedure non standard agendo sui tre vettori.

ColdCapture è il vettore che indica al sistema l'indirizzo in RAM di una routine di cold-capture non standard. Il sistema accede a questo vettore proprio all'inizio della procedura di attivazione della macchina avviata conseguentemente a un reset da tastiera (combinazione di tasti Control/Amiga-Sinistro/Amiga-Destro), oppure all'accensione della macchina. Inserendo l'indirizzo di una routine in questo vettore, si può intervenire sulla procedura di attivazione nel suo punto d'inizio, spodestando la procedura predefinita internamente.

Dopo qualche tempo, sempre durante la procedura di attivazione, il sistema controlla automaticamente il valore contenuto nel vettore CoolCapture. Se risulta diverso da zero, il sistema continua la procedura di attivazione cedendo il controllo all'indirizzo indicato, altrimenti la sequenza continua senza interruzioni. Inserendo un indirizzo nel vettore CoolCapture, si riesce a intervenire sulla procedura di attivazione in un suo particolare punto intermedio.

Nella terza fase, dopo un ulteriore lasso di tempo, il sistema accede al vettore WarmCapture, e cede il controllo all'indirizzo in esso contenuto se è diverso da zero, ignorando così la terza fase standard. Inserendosi a questo livello, un task modifica solo l'ultima fase della procedura di attivazione standard.

Dove si trova lo stack di sistema? SysStkUpper contiene l'indirizzo superiore del supervisor mode stack del 68000, il cosiddetto stack di sistema dell'Amiga (per indirizzo superiore s'intende l'indirizzo della cima dello stack). Si noti che il parametro SysStkUpper della struttura ExecBase è la controparte del parametro tc_SPUpper contenuto nella struttura Task, ovvero il puntatore all'indirizzo superiore dello user mode stack associato al corrispondente task.

Benché ogni task del sistema possieda un suo personale stack, il sistema per le proprie operazioni interne fa uso solo del supervisor mode stack. In una macchina dotata di 512K, in genere, l'indirizzo contenuto in questo parametro è 0x00080000 (decimale 524288), come si può facilmente verificare ricorrendo al *Source Level Debugger* della Manx, e indica al sistema di allocare la cima dello stack di sistema all'indirizzo assoluto più grande nella RAM disponibile. Il supervisor mode stack è l'area di memoria RAM che il sistema utilizza per salvare e ripristinare i valori contenuti nei registri della CPU che caratterizzano il sistema in particolari momen-

ti (per esempio, quando si verifica un interrupt hardware o una trap), ed è inoltre impiegato per immagazzinare gli indirizzi di ritorno quando all'interno del sistema vengono effettuate chiamate alle subroutine interne mentre è attivato il supervisor mode.

Benché non sia frequente, un task definito dal programmatore può imporre al sistema di entrare nel supervisor mode chiamando la funzione SuperState della libreria Exec. Se si decide di effettuare questa inconsueta operazione, al posto del supervisor mode stack verrà usato lo user mode stack associato al task, e si deve quindi tener conto che potrebbe non essere abbastanza capiente durante l'esecuzione delle routine di interrupt hardware o di trap.

In particolare, se si verifica l'esecuzione ravvicinata di molte routine di interrupt nidificate (cioè, se si verificano in rapida successione molti interrupt hardware, per esempio da disco), è possibile che una routine di interrupt venga mandata in esecuzione prima che la precedente abbia completato la sua esecuzione, e il numero di parametri che a mano a mano verrebbero memorizzati nello stack in un caso di questo tipo potrebbe diventare troppo elevato. Inoltre, lo stack di sistema dev'essere abbastanza grande da risultare sufficientemente capiente anche per tutti gli indirizzi di ritorno delle subroutine associate alle chiamate che vengono effettuate nel corso della routine di interrupt hardware o di trap.

Per esempio, se il programma chiama una routine di trap software tramite l'istruzione Assembly TRAP #N, e questa routine chiama a sua volta alcune routine della libreria Exec, il supervisor mode stack dev'essere abbastanza grande da contenere tutti gli indirizzi di ritorno di tali routine, cosicché il controllo possa tornare alla routine di trap non appena ciascuna funzione è giunta al termine della sua esecuzione (si ricordi che il sistema entra sempre nel supervisor mode quando esegue una routine di trap software o hardware).

SysStkLower contiene l'indirizzo inferiore del supervisor mode stack del 68000 (per indirizzo inferiore s'intende l'indirizzo del fondo dello stack). In una macchina dotata di 512K questo indirizzo è in genere 0x0007E800. La dimensione dello stack di sistema viene stabilita durante la procedura di attivazione dalle routine del *Kickstart*, ed è abbastanza grande da poter gestire tutte le routine di interrupt ad alta priorità dell'Exec, tenendo conto della possibilità di nidificazione e delle chiamate alle subroutine, e tutte le routine di interrupt hardware e di trap.

La grandezza dello stack di sistema può essere calcolata sottraendo SysStkLower da SysStkUpper, e in una macchina dotata di 512K risulta di 6144 byte, ovvero superiore agli user mode stack associati ai task, che non superano in genere i 2000 byte.

Quanta memoria chip è presente nel sistema? MaxLocMem contiene l'ultimo valore calcolato della massima memoria chip RAM disponibile nel sistema. Per l'Amiga 500, 2000 e 1000 con 512K, questo parametro contiene il valore 512K o, più precisamente, 524288 byte. Questo particolare valore indica che nel sistema sono presenti 512K di memoria chip RAM. Si veda anche il parametro MaxExtMem, esaminato più avanti. Il sistema è progettato in modo tale che una particolare routine della libreria Exec accede, durante la procedura di attivazione, ad alcuni segnali hardware provenienti dalle schede di memoria per valutare la quantità di memoria chip RAM presente e aggiornare conseguentemente questo parametro della struttura ExecBase. Si noti che questo parametro è per definizione un parametro a sola lettura, quindi un task definito dal programmatore non deve cambiarlo.

Come si effettua il debug? DebugEntry è il puntatore all'entry point del cosiddetto ROMWack Debugger della libreria Exec, un insieme di routine di debug contenute in ROM, alle quali viene ceduto il controllo tramite la funzione Debug() della libreria Exec. Ricorrendo al *Source Level Debugger* della Manx si può verificare che questo parametro vale 0x00FC237E nella versione 1.2.1 del software sistema. La libreria Debug contiene in ROM diverse funzioni che un programma può chiamare per effettuare il debug. È importante notare che l'uso del ROMWack Debugger richiede un computer connesso all'Amiga attraverso la porta seriale, in modo da visualizzare i risultati delle chiamate alle funzioni di debug sul monitor del computer remoto (nel trasferimento dei dati attraverso la linea seriale, la velocità di trasmissione è 9600 baud). Nella versione 1.3 del sistema operativo dell'Amiga c'è anche una libreria di debug (Ddebug) che prevede l'uso della porta parallela per permettere al programmatore di accedere ai risultati del debug tramite una stampante collegata appunto alla porta parallela. Dal momento che quasi tutti hanno una stampante, mentre è più raro che si possieda un computer remoto da interfacciare, questa aggiunta è utilissima in sede di debug. La libreria Ddebug prevede gli stessi meccanismi offerti dal debug che usa la porta seriale, e ricorre alle stesse funzioni anche se con alcune piccole variazioni nei nomi: kprintf() anziché dprintf(), per esempio. In altre parole, l'unica cosa che cambia è che i risultati vengono visualizzati tramite stampante anziché sul video del computer remoto.

Se un programmatore sa già come servirsi delle routine del ROMWack Debugger, può leggere il valore del parametro DebugEntry (che può variare da versione a versione) per accedere alle routine di debug. Più precisamente, può far eseguire alla sua applicazione un salto all'entry point del ROMWack Debugger, visualizzare su monitor o su stampante i valori delle variabili del suo programma, e vedere

al tempo stesso i risultati prodotti dall'esecuzione del programma sul monitor standard dell'Amiga. Si tratta di una procedura di debug alternativa, benché più macchinosa e più restrittiva del consueto ricorso al *Source Level Debugger* della Manx (che descriveremo dettagliatamente in uno dei prossimi articoli).

DebugData è il puntatore a una locazione nella memoria RAM contenente particolari dati di debug creati dal programmatore per sostituire quelli standard. I dati di debug vengono utilizzati dalle routine del ROMWack Debugger. Questo parametro di solito contiene il valore zero, per indicare alle routine di debug interne che non è stato preparato nessun particolare dato di debug, e che quindi devono utilizzare i dati contenuti in ROM. In pratica questo valore è diverso da zero soltanto se qualcuno ha modificato il sistema per collocare dati destinati al ROMWack Debugger in una specifica locazione RAM prima di attivarlo, in modo da personalizzare le procedure di debug.

Come sono gestiti i messaggi di Guru Meditation? AlertData, quando è diverso da zero, contiene l'indirizzo dei dati di alert che sostituiscono quelli standard previsti dal sistema Exec. Viene usato dalle routine interne di Guru Meditation dell'Exec per impostare e visualizzare messaggi non standard quando avviene un crash del sistema. In genere contiene il valore zero, per indicare che i dati di sistema che caratterizzano i messaggi di alert non sono stati scavalcati da nuovi dati impostati dal programmatore in RAM.

Molti ritengono che i messaggi di Guru Meditation non siano affatto chiari per l'utente medio. Anche i progettisti dell'Amiga non erano del tutto convinti del sistema scelto per definire e visualizzare i messaggi d'errore, ma mancava il tempo per modificare radicalmente il sistema e arrivare a una presentazione visiva più simile, per esempio, a quella del Macintosh. Per queste ragioni, i programmatori che non sono soddisfatti dei normali messaggi di Guru Meditation possono alterare AlertData per definire nuovi messaggi d'allarme più comprensibili e un nuovo sistema di visualizzazione.

Quanta memoria RAM estesa è presente? MaxExtMem contiene l'indirizzo più alto nella memoria RAM estesa disponibile nel sistema. Questo valore rappresenta la quantità di memoria presente che va oltre a quella chiamata chip, la cui quantità è contenuta nel già descritto parametro MaxLocMem. In genere tutta la memoria estesa viene considerata memoria MEMFAST, cioè fast RAM (si ricordi che i chip custom dell'Amiga non possono accedere alla memoria fast). Le routine interne dell'Exec impostano questo valore durante la procedura di attivazione della macchina. Se si aggiungono una o più schede di memoria al sistema per aumentarne le capacità di memoria al di sopra dei 512K, l'hardware ne rileva la presenza e la segnala alle routine interne perché calcolino il nuovo indirizzo da scrivere in MaxExtMem, mentre se non è stata installata nessuna espansione RAM, contiene il valore zero. Se un programma ha bisogno per funzionare di una quantità di memoria RAM superiore a quella chip, deve innanzi tutto andare a leggere questo parametro per assicurarsi che la memoria complessiva della macchina sia sufficiente. Nel caso che non sia così, il programma può visualizzare un requester tramite Intuition e avvisare l'utente del problema. Questo controllo viene svolto da quasi tutti i programmi in commercio che richiedono una quantità di memoria RAM superiore a 512K.

L'attivazione del sistema è avvenuta correttamente? ChkSum contiene il valore di checksum ottenuto dal sistema effettuando il controllo su alcuni dei precedenti 13 parametri (o su tutti) della struttura ExecBase, da SoftVer a MaxExtMem. La maggior parte dei valori presenti in questi parametri dipendono da dati contenuti in ROM o dalla specifica configurazione della macchina, e quindi non dovrebbero cambiare una volta che la ROM e la configurazione hardware sono state completamente definite. Il sistema si serve di questo parametro per verificare se la sua integrità – ovvero i valori contenuti nelle locazioni di questi 13 parametri – è stata violata durante la procedura di attivazione. In questo modo l'utente può essere avvertito di un possibile problema prima di mandare in esecuzione un programma. Anche qui ci troviamo di fronte a un parametro a sola lettura, che non dovrebbe essere alterato dai task definiti dal programmatore.

I prossimi 16 parametri riguardano gli interrupt hardware dell'Amiga. Si veda l'articolo pubblicato sul numero 5/88 di *Commodore Gazette* per informazioni su come si verificano gli interrupt hardware nel corso delle operazioni di sistema, e su come operano le relative routine mentre gli eventi del mondo esterno si intromettono nel flusso operativo dell'Amiga.

Dove sono i vettori di interrupt hardware? IntVects[16] è una raccolta di 16 strutture di tipo IntVector. Ciascuna struttura IntVector rappresenta uno dei 16 interrupt hardware possibili nel sistema dell'Amiga. Anche queste voci riflettono soltanto variazioni del sistema che scavalcano le condizioni precostruite in ROM; in altre parole, sono diverse da zero soltanto se le routine di interrupt hardware contenute in ROM sono state scavalcate da nuove routine impostate da uno o più task (facendo uso della funzione SetIntVector).

La struttura IntVector è definita nel file INCLUDE interrupts.h. Ogni struttura IntVector possiede un puntatore a una routine di interrupt hardware predefinita, il parametro iv_Code(), e un puntatore a un indirizzo per i dati utilizzati dalla routine, il parametro iv_Data. Questi due parametri indicano alle routine della libreria Exec dove

sono collocati i codici e i dati necessari per eseguire le operazioni conseguenti a un interrupt hardware. A meno che un task non esegua una chiamata alla funzione SetIntVector, questi due parametri delle sotto-strutture IntVector appartenenti alla struttura ExecBase valgono zero per ciascuno dei 16 interrupt, indicando così che tutti gli interrupt hardware verranno gestiti tramite le routine predefinite in ROM. Proprio come avviene per il parametro LibNode della sotto-struttura Library, queste 16 voci dell'array IntVector rappresentano sotto-strutture, e non puntatori.

Ecco l'ordine in cui sono conservate le 16 sotto-strutture IntVector.

– Interrupt per segnalare che il buffer di trasmissione della porta seriale è vuoto. Questo interrupt si verifica quando il buffer della porta seriale ha trasmesso il suo ultimo byte attraverso la porta seriale a un dispositivo seriale esterno.
– Interrupt per segnalare che il trasferimento di un blocco del disco è stato completato. Questo interrupt si verifica quando, in un'operazione di lettura o scrittura, è stato trasferito l'ultimo byte di un blocco di grandezza prestabilita.
– Interrupt software del sistema Exec. Questo interrupt si verifica quando dev'essere eseguita una routine di interrupt software (exception). Può essere causato tanto dalla funzione Cause quanto dal segnale di una message port.
– Interrupt delle porte di I/O (tastiera, clock, periferiche). Questo interrupt si verifica quando un dispositivo hardware come la tastiera, il clock e così via, richiede l'attenzione del sistema.
– Uno degli interrupt del coprocessore Copper.
– Interrupt relativo alla scansione verticale dello schermo effettuata dal pennello elettronico. Questo interrupt si verifica quando il pennello elettronico, dopo aver scandito un intero quadro del video, si spegne e torna all'angolo superiore sinistro per iniziare un nuovo quadro. Questa operazione si verifica 60 volte al secondo per le macchine americane (sistema NTSC) e 50 volte al secondo per le macchine europee (sistema PAL).
– Interrupt del coprocessore Blitter.
– Quattro interrupt relativi ai quattro canali audio.
– Interrupt per segnalare che il buffer di ricezione della porta seriale è pieno. Questo interrupt si verifica quando il buffer di ricezione della porta seriale è stato completamente riempito (in genere quando contiene 512 byte).
– Interrupt per segnalare la ricezione di un byte da disco. Questo interrupt si verifica ogni volta che viene completato il trasferimento di un byte da disco (in lettura o scrittura).
– Interrupt del connettore d'espansione esterno. Questo interrupt si verifica quando un dispositivo hardware esterno invia un segnale di interrupt al 68000.
– Interrupt relativo al coprocessore Copper.
– Interrupt non mascherabile del 68000. Questo interrupt si verifica quando un dispositivo hardware esterno invia un segnale di interrupt al 68000. È l'unico interrupt dell'Amiga che non può essere ignorato (mascherato) variando la priorità di interrupt del 68000.

Si veda il file INCLUDE execbase.i per una definizione più precisa di queste sotto-strutture IntVector presenti nella struttura ExecBase.

Una verifica pratica

Per verificare il contenuto dei parametri della struttura ExecBase presente nel proprio Amiga, proponiamo un semplice programma in C. La sua funzione, main(), visualizza i valori contenuti in alcuni dei parametri che abbiamo illustrato nell'articolo. I valori vengono presentati in esadecimale, ma non è difficile alterare gli argomenti delle funzioni printf() perché vengano visualizzati in esadecimale.

Se si dispone del compilatore C della Manx, per compilare questo programma, che supponiamo si chiami *exec.c*, si deve anzitutto impartire il seguente comando:

```
CC exec
```

Se la compilazione del programma ha avuto successo, si procede alla fase di link utilizzando il seguente comando:

```
LN exec -lc
```

A questo punto il programma è stato compilato, e può quindi essere eseguito. Eccone ora il codice sorgente.

```
#include <exec/execbase.h>

struct ExecBase *SysBase;

main()

{
 printf("SoftVer         %lx\n",SysBase->SoftVer);
 printf("ColdCapture     %lx\n",SysBase->ColdCapture);
 printf("CoolCapture     %lx\n",SysBase->CoolCapture);
 printf("WarmCapture     %lx\n",SysBase->WarmCapture);
 printf("SysStkUpper     %lx\n",SysBase->SysStkUpper);
 printf("SysStkLower     %lx\n",SysBase->SysStkLower);
 printf("MaxLocMem       %lx\n",SysBase->MaxLocMem);
 printf("MaxExtMem       %lx\n",SysBase->MaxExtMem);
 printf("DebugEntry      %lx\n",SysBase->DebugEntry);
}
```

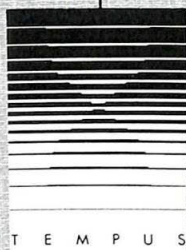
Argomenti futuri

Nel prossimo articolo completeremo la trattazione della struttura ExecBase, discutendo i rimanenti 34 parametri (da ThisTask a ExecBaseNewReserved). Verranno discusse le ultime variabili statiche, ma l'attenzione sarà rivolta soprattutto ai cosiddetti parametri dinamici, ovvero quelli che dopo l'attivazione del sistema variano ripetutamente. Il programma che abbiamo presentato si rivelerà utile anche per verificare il contenuto di questi parametri.

■

ESTRATTO DAL

CATALOGO LIBRI 1988/89

COLLANA TEMPUS scienza e tecnologia

La storia, l'impatto sociale e il futuro della scienza e dell'alta tecnologia

La IHT Gruppo Editoriale inaugura il suo terzo anno di vita dando l'avvio a una nuova collana che si occuperà di un settore ancora tutto da esplorare: quello della ricerca scientifica e tecnologica.

Con questa collana cominciano a precisarsi alcuni obiettivi della nostra casa editrice, che non intende limitarsi a pubblicare testi d'informatica – sia pure estremamente curati e scelti tra i più significativi del mercato mondiale – ma vuole offrire al pubblico un catalogo quanto più vario possibile che, privilegiando sempre il punto di vista tecnologico, possa aiutare i suoi lettori a tenersi aggiornati con ogni aspetto del mondo moderno.

Gli argomenti della collana, le sue tematiche, le sue linee d'interesse diventano sempre più evidenti leggendo la descrizione dei volumi presentati in questo catalogo. Non ci sono limiti "settoriali" alle esplorazioni di TEMPUS: vengono proposti testi sull'intelligenza artificiale, sull'inserimento dei nuovi sistemi "intelligenti" nella società di oggi, sui progressi compiuti nel campo della genetica, sull'evoluzione dell'economia moderna, sulle moderne teorie dell'evoluzione dell'universo, sui sistemi d'arma del futuro, sulla possibilità di essere ancora un "inventore" in un mondo in cui sembra esistere solo la ricerca di gruppo...

È una serie che si rivolge a tutti i lettori aperti ai nuovi stimoli della scienza, una scienza presente nella vita di ogni giorno, non solo nei laboratori e nelle università. E proprio per questo i libri che vengono selezionati per il marchio TEMPUS privilegiano la facilità di lettura, pur senza venire mai meno al rigore dei contenuti e alla completezza dell'informazione.

Noi non abbiamo mai creduto che solo chi è in possesso di un linguaggio matematico adeguato debba avere l'opportunità di affrontare il mondo della scienza. I linguaggi ermetici sono necessari soltanto per argomenti strettamente specialistici, e troppo spesso si leggono libri inutilmente ostici. Non abbiamo alcuna intenzione di battere ancora questa strada.

La collana TEMPUS propone libri stimolanti e vivaci, che possono aiutare l'uomo di oggi a entrare in contatto con la realtà affascinante in cui vivrà l'uomo di domani.

LA MACCHINA E LA MENTE

Alla scoperta della nuova scienza dell'intelligenza artificiale

George Johnson

È possibile creare una macchina che comprenda il linguaggio umano? Che faccia scoperte originali e sappia imparare dai propri errori? Quanto vicini siamo a risultati di questo tipo? Quanto lontani potremo spingerci?

La macchina e la mente offre un ritratto vivido e profondo di una scienza che sta nascendo, una relazione dalle più lontane frontiere a cui si sia spinto l'uomo: l'intelligenza artificiale (AI). È l'irresistibile storia di un'impresa che coinvolge intrallazzi politici, speculazioni selvagge, ostacoli impossibili, grandi trionfi e abbaglianti visioni.

In questo libro facciamo conoscenza con i padri dell'intelligenza artificiale: Marvin Minsky, Roger Schank, Douglas Hofstadter e Edward Feigenbaum. Dalle loro dirette esperienze scopriamo a che punto è arrivata la progettazione di una macchina che comprende il linguaggio umano e su quali teorie scientifiche è basata.

Accanto alla storia puramente tecnica dell'intelligenza artificiale, inoltre, *La macchina e la mente* prende in considerazione anche le implicazioni etiche e filosofiche di una linea di ricerca tanto delicata. Il cervello è solo un computer, che elabora una serie di programmi chiamati "mente"? Ha senso pensare di creare sistemi più intelligenti dei loro stessi creatori? È pericoloso lasciare che siano le macchine a prendere certe decisioni? Se daremo la capacità di ragionare alle macchine, daremo loro anche l'autocoscienza?

In 400 pagine, questo libro offre molte risposte, e moltissimi interrogativi, dal momento che oggi siamo arrivati al punto di poter parlare di "scienza dell'intelligenza artificiale" (e di raccontarne la storia), ma intravediamo un cammino ancora molto lungo. In un certo senso ci sentiamo sul confine tra il naturale e l'artificiale... ma non siamo del tutto certi che questo confine esista davvero.

400 pagine - 16 × 23,5 cm - rilegato con copertina rigida - ISBN 88-7803-012-0 - L. 42.000
Data di pubblicazione: ottobre/dicembre 1988

INVENTORI DEL NOSTRO TEMPO

Interviste con 16 famosi inventori americani

Kenneth A. Brown

Inventori del nostro tempo è una raccolta di 16 interviste con alcuni dei maggiori inventori di questo secolo. Da professionisti della ricerca come Maxime Faget della NASA, Harold Rosen della Hughes Aircraft e Bob Gundlach della Xerox, fino a imprenditori indipendenti come Stanford Ovshinsky e Raymond Kurzweil, esperto di intelligenza artificiale.

Le loro realizzazioni – il laser, il microprocessore, il pacemaker, il computer Apple II e molte altre – sono spesso coraggiose e originali, e sempre brillanti.

A poco a poco, attraverso le diverse storie dei personaggi intervistati, cominciamo a intuire quali regole comandano i meccanismi della dote che chiamiamo "creatività". Ma l'economia capitalistica rende la vita difficile allo spirito creativo, tanto che molti, già oggi, pensano che gli inventori siano una razza estinta.

Invece, come testimonia questo libro, si tratta di un'ipotesi assolutamente infondata... almeno per ora.

400 pagine - 16 × 23,5 cm - rilegato con copertina rigida
ISBN 88-7803-010-4 - L. 42.000
Data di pubblicazione: ottobre/dicembre 1988

COMPUTER IN GUERRA: FUNZIONERANNO?

I rischi e le potenzialità delle nuove tecnologie militari

David Bellin e Gary Chapman

Stiamo per caso affidando la sicurezza del mondo a tecnologie troppo avveniristiche? Questa è la domanda a cui si sforza di rispondere *Computer in guerra: funzioneranno?*

Questo volume presenta gli elementi scientifici concreti necessari per crearsi un'opinione al riguardo, e riserva un ampio spazio alle analisi dei maggiori esperti. Il risultato è un libro non tecnico, di facile lettura e molto stimolante.

Tra i principali argomenti affrontati da *Computer in guerra: funzioneranno?* troviamo lo "scudo stellare", il programma "strategic computing", le applicazioni dell'intelligenza artificiale alla tecnologia militare e così via.

350 pagine - 16 × 23,5 cm - rilegato con copertina rigida
ISBN 88-7803-011-2 - L. 39.000
Data di pubblicazione: ottobre/dicembre 1988

Distributore nazionale

Messaggerie Libri
Via Giulio Carcano, 32 - 20141 Milano
Tel. 02/8438141 - 8467341 - Telex 310672 MESSIT I

Filiali

BARI: Via Caduti del Lavoro, 2/A - 70126 Bari - tel. 080/483880/1 - province: Bari, Brindisi, Foggia, Lecce, Matera, Taranto - **BOLOGNA:** Via del Tuscolano, 3/5 - 40128 Bologna - tel. 051/324610-323768 - province: Bologna, Ancona, Ascoli Piceno, Ferrara, Forlì, Macerata, Mantova, Modena, Parma, Pesaro, Urbino, Ravenna, Reggio Emilia - **CAGLIARI:** Via del Commercio, 27 - 09100 Cagliari - tel. 070/287902 - province: Cagliari, Nuoro, Oristano, Sassari - **FIRENZE:** Via Francesco Baracca, 183 - 50127 Firenze - tel. 055/430223 - 432312 - province: Firenze, Arezzo, Grosseto, Livorno, Lucca, Massa Carrara, Perugia, Pisa, Pistoia, Siena - **GENOVA:** Via Gelasio Adamoli, 261/263 - 16141 Genova - tel. 010/869722-869769 - province: Genova, Imperia, La Spezia, Savona - **MILANO:** Via Volta, 13 - 20089 Rozzano - tel. 02/8240951/2/3 - 8258252 - province: Milano, Bergamo, Brescia, Como, Cremona, Novara, Pavia, Piacenza, Sondrio, Varese - **NAPOLI:** Via Nazionale delle Puglie, km. 36.150 - 80013 Casalnuovo - tel. 081/8423233 - 8423246 - province: Napoli, Avellino, Benevento, Campobasso, Caserta, Catanzaro, Cosenza, Isernia, Potenza, Salerno - **PADOVA:** Via Danieletti, 41 - 35100 Padova - tel. 049/609711 - province: Padova, Belluno, Bolzano, Gorizia, Pordenone, Rovigo, Trento, Treviso, Trieste, Udine, Venezia, Verona, Vicenza - **PALERMO:** Via Savonarola, 9 - 90135 Palermo - tel. 091/404289 - 403165 - province: Palermo, Agrigento, Caltanissetta, Catania, Enna, Messina, Ragusa, Reggio Calabria, Siracusa, Trapani - **ROMA:** Via Valtellina, 79 - 00151 Roma - tel. 06/538941/2 - 5313797 - province: Roma, Chieti, Frosinone, L'Aquila, Latina, Pescara, Rieti, Teramo, Terni, Viterbo - **TORINO:** Corso Peschiera, 321/0 - 10141 Torino - tel. 011/728073 - 726746 - province: Torino, Alessandria, Aosta, Asti, Cuneo, Vercelli.

Come ricevere i libri IHT

Librerie, cartolibrerie, grossisti di libri, computer shop, sono serviti dal nostro distributore nazionale Messaggerie Libri. Le richieste possono essere fatte per posta o telefonicamente alla filiale di competenza. I privati possono prenotare i libri IHT tramite l'apposito tagliando pubblicato sulla rivista Commodore Gazette, oppure ordinarli telefonicamente. L'indirizzo a cui rivolgersi è:

IHT Gruppo Editoriale
Via Monte Napoleone, 9 - 20121 Milano
Tel. 02/794181 - 799492 - 792612 - 794122
Telex 334261 IHT I - Fax 02/784021

I prezzi possono essere modificati senza preavviso. I pagamenti possono essere effettuati in contrassegno, con addebito delle relative spese postali, oppure anticipati a mezzo assegno o vaglia postale intestato alla IHT Gruppo Editoriale. In quest'ultimo caso, è necessario allegare all'ordine fotocopia della ricevuta, e telefonare per informarsi riguardo all'importo delle spese postali.

NORME PER I COLLABORATORI

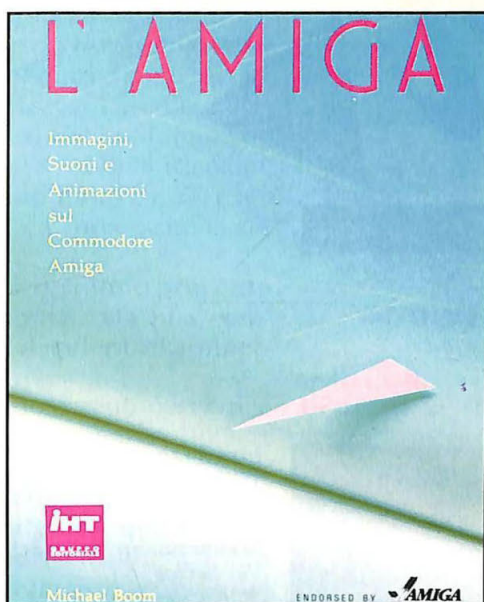
Siete interessati a collaborare con la nostra casa editrice in qualità di autori, articolisti o traduttori?
In linea di massima prendiamo in considerazione ogni proposta, ma il nostro lavoro verrà semplificato se seguirete alcune semplici norme:

autori
- inviare curriculum accademico e professionale, e una nota biografica
- piano dell'opera proposta, con dettagliata descrizione della struttura e dei contenuti
- dimensioni approssimative dell'opera e data di consegna

articolisti
- inviare curriculum accademico e professionale, e una breve nota biografica
- breve descrizione dell'articolo, della serie di articoli, o del listato proposto
- dimensioni approssimative dell'articolo (in cartelle)
- eventuale disponibilità a una collaborazione continuativa
- nel caso di un listato, si ricordi che l'autore deve scrivere anche un articolo di presentazione

traduttori
- inviare curriculum accademico e professionale, e una breve nota biografica, specificando prima, seconda e (eventualmente) terza lingua
- specificare se la proposta riguarda la collana Informatica (traduzioni di tipo tecnico) o le collane Cinema e Tempus

Ogni proposta va inviata a:
IHT Gruppo Editoriale
Servizio Collaboratori
Via Monte Napoleone, 9 - 20121 Milano
Tel. 02/794181 - 799492 - 792612 - 794122

Un computer chiamato AMIGA

L'AMIGA

Come il Commodore Amiga ha cambiato il mondo dei computer, così il volume *L'Amiga* proietterà la vostra immaginazione lungo nuovi orizzonti di creatività. Preparatevi a ottenere stupefacenti risultati dal vostro computer: sofisticate immagini video, suoni e musica, sequenze animate da registrare su videocassetta, e molto di più. In questo volume troverete inoltre consigli utili per ottenere il meglio dall'Amiga Basic e da prodotti software eccezionali come Deluxe Paint, Deluxe Music e Deluxe Video.

416 pagine, oltre 100 illustrazioni, L. 60.000

IL MANUALE DELL'AMIGADOS

Questo volume è l'unica documentazione ufficiale realizzata dalla Commodore sul sistema operativo dei computer Amiga. Contiene tre libri (Il Manuale per l'utente, Il Manuale per il programmatore e Il Manuale di riferimento tecnico) che costituiscono nel loro complesso la guida più completa per ogni utente dell'Amiga, dal principiante che vuole usare i comandi dell'AmigaDOS (che non sono spiegati dall'opuscolo in dotazione al computer), fino al programmatore evoluto che troverà utili informazioni per programmare in C e in Assembly.

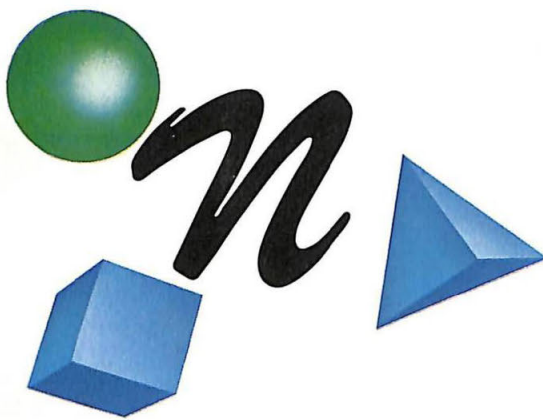
376 pagine, L. 60.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127

IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I

Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

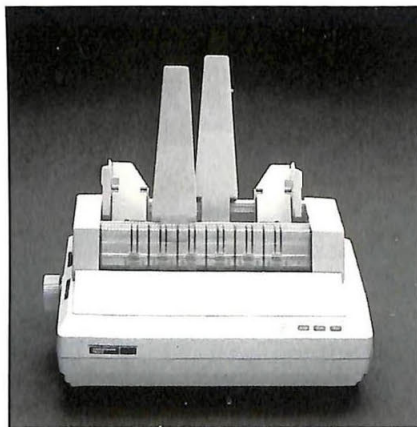
COMMODORE NEWS

NOVITÀ HARD E SOFTWARE DALL'ITALIA E DAL MONDO

ITALIA

UNA NUOVA MANNESMANN TALLY

Anche se chi acquista una stampante a matrice di punti da nove aghi non pretende in genere una stampa di elevata qualità né una produttività particolarmente alta, la Mannesmann Tally promette – con il nuovo modello MT81 – una tecnologia raffinata e una buona affidabilità.

Le sue caratteristiche tecniche sono: stampa bidirezionale a 130 caratteri al secondo in alta velocità, stampa unidirezionale a 24 caratteri al secondo in alta definizione, densità di punti in grafica compresa fra i 60 e i 240 punti per pollice, con una densità di linea di 6/8/12 per pollice.

Per quel che riguarda il trattamento della carta, la stampante MT81 gestisce i fogli singoli con il trascinamento a frizione e i moduli continui con i trattori di spinta; sono inoltre disponibili in opzione un caricatore di fogli singoli e una vasca con una capacità di cento fogli per la stampa

non assistita dei documenti.

Pur proposta a un costo contenuto, la nuova stampante non rinuncia a offrire procedure particolarmente semplici per la regolazione dei parametri di stampa, attraverso un pannello frontale dotato di tre selettori e di quattro led che permettono un'impostazione rapidissima e controllano l'eventuale stato di errore della macchina. La MT81 dispone di un'interfaccia parallela standard. Sono anche disponibili le interfacce seriale e Commodore (per C-64/128); la comunicazione con i più comuni sistemi presenti sul mercato è resa possibile dalle emulazioni IBM Proprinter II, Epson FX 80 e Commodore.

Mannesmann Tally Srl
Via Borsini, 6
20094 Corsico (MI)
(tel. 02/4502850)

CONCORSO BIT MOVIE

È stato indetto il concorso "Bit Movie '89", dedicato all'animazione grafica in tempo reale su personal computer. Le migliori opere concorrenti verranno esposte in occasione dell'omonima mostra, organizzata dal Circolo Arci Ratataplan di Riccione, che si terrà nei giorni di sabato, domenica e lunedì di Pasqua 1989, nei locali del Centro della Pesa in Corso Fratelli Cervi a Riccione.

Regolamento.

1) Il Circolo Arci Ratataplan in collaborazione con il Comune di Riccione, organizza il secondo concorso per animazione grafica in tempo reale.

2) Sono ammesse tutte le opere realizzate su personal computer ed elaborate in tempo reale.

3) Possono essere utilizzati tutti i programmi grafici esistenti, ma non oggetti o sfondi tratti da librerie di pubblico dominio.

4) Ogni concorrente può presentare al massimo quattro opere.

5) I lavori dovranno essere registrati su floppy disk o su hard disk.

6) Le opere dovranno pervenire, franco di ogni spesa, al seguente indirizzo: Circolo Arci Ratataplan c/o Carlo Mainardi – Via Bologna, 13 – 47036 Riccione (FO). Dovranno inoltre essere accompagnate da una scheda di commento.

7) Le opere verranno preventivamente esaminate da una commissione che deciderà se ammetterle al concorso.

8) I lavori dovranno pervenire entro il giorno 15 marzo 1989.

9) Le opere saranno giudicate dal pubblico con apposite schede voto e da una giuria qualificata.

10) Il supporto magnetico sarà restituito all'autore al termine del concorso.

Premi:

All'autore dell'opera prima classificata L. 500.000.

All'autore dell'opera seconda classificata L. 300.000.

All'autore dell'opera terza classificata L. 200.000.

Circolo Arci Ratataplan
Per informazioni telefonare a:
Carlo Mainardi (0541/42878)
Stefano Leardini (0541/377388)

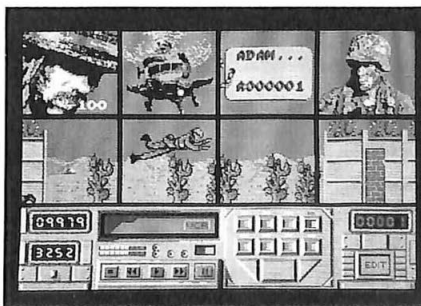
FLICKER MASTER

Finalmente un metodo per tenere sotto controllo il flickering del monitor in modo interlace! Lo schermo *Flicker Master* riesce a migliorare il contrasto a tutte le risoluzioni, riduce lo sforzo a cui viene sottoposta la vista, fornisce una speciale "pigmentazione neutra" per diminuire l'alterazione dei colori e soprattutto elimina quasi completamente il flickering (sfarfallio) in modo interlace. È stato appositamente studiato per essere applicato sui monitor Amiga 1080 e 2002 (e in genere per i monitor da 10,5 o 13,5 pollici). Il prezzo per il mercato italiano è di L. 29.000 (Iva compresa).

Lago snc
Via Napoleona, 16
22100 Como
(tel. 031/300174)

ESTERO

INFOGRAMES NEWS

In qualche sperduto punto dell'Europa si trova un centro di addestramento segreto: la scuola più dura del mondo. Per incontrare i soldati di questa scuola, e mettere alla prova coraggio, forza e saldezza di nervi, basta caricare nel proprio Amiga *Action Service*, l'ultimo prodotto distribuito dalla Infogrames.

Il progetto legato a questo programma è molto ambizioso. Non si tratta infatti di un gioco a se stante, ma di una serie il cui nucleo è appunto la scuola di combattimento a cui abbiamo accennato: ogni nuovo episodio costituirà una diversa missione, e come in tutti i giochi d'azione bisognerà mantenersi in perfetta forma e migliorare ogni volta le proprie prestazioni per dimostrarsi all'altezza. Il secondo episodio, *Action Service 2*, è già in fase di realizzazione e sarà disponibile entro breve tempo.

Per quanto riguarda l'aspetto tecnico, invece, diremo soltanto che questo programma propone un "construction set" tanto ricco che sulle prime era stata ventilata l'idea di pubblicarlo a parte. Ma i distributori americani non hanno accettato questa soluzione, e non è sembrato corretto penalizzare il mercato europeo proponendo un prodotto meno completo di quello che sarebbe stato disponibile negli Stati Uniti.

La raffinatezza della programmazione è confermata anche dalla possibilità di far compiere al proprio personaggio molte azioni contemporaneamente: si può farlo nuotare, saltare e nello stesso istante lanciare una granata...

Infogrames
84, Rue de 1er Mars 1943
69628 Villeurbanne Cedex (France)

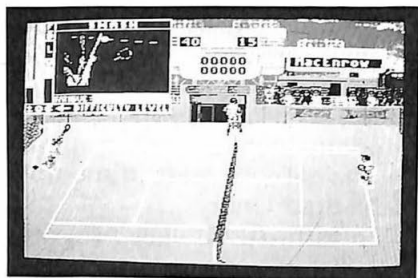
JOYSTICK SUNCOM

La Suncom, consociata della Microprose, annuncia l'imminente introduzione di due nuovi modelli di joystick, il Tac 30 e il Tac 50, che saranno disponibili per il mercato europeo quattro mesi prima che negli Stati Uniti. I due nuovi modelli sono la versione riveduta e corretta dei joystick Tac 3 e Tac 5 (che continuano a essere disponibili). Tra i miglioramenti introdotti segnaliamo il funzionamento tramite microswitch, il nuovo disegno "aircraft style", e una novità assoluta: un auto-fire variabile.

Come per tutti i prodotti Suncom, i due joystick hanno una garanzia di due anni. Il prezzo di vendita è fissato rispettivamente in 13,99 e 14,99 dollari.

Microprose Software Ldt
2, Market Place, Tetbury
Glos GL8 8DA, England
(tel. 0044/0666/54331)

SIMULAZIONI SPORTIVE

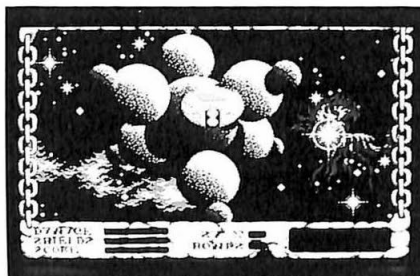
La Accolade ha annunciato nuove proposte software, tra le quali spiccano ben tre simulazioni sportive.

Serve & Volley è dedicata al tennis. Una simulazione che unisce strategia e azione, mettendo l'accento sul corretto posizionamento della palla, e sull'accuratezza nello scegliere i tempi e la forza per il tiro. Tre diversi campi di gioco e tre crescenti livelli di difficoltà ne fanno un gioco abbastanza vario, ma la novità di maggior rilievo è la nuova tecnica d'animazione strobe-o-strobe, che rende particolarmente realistici colpi e servizi.

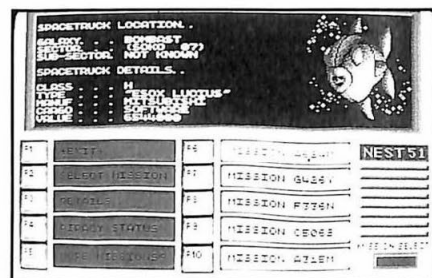
Fast Break è una simulazione di pallacanestro, in cui si può scegliere se giocare in attacco o in difesa. Due giocatori competono testa a testa (ma si può giocare anche contro il computer) in una partita in cui valgono tutte le regole del basket professionistico.

T.K.O., ovvero "KO tecnico", è naturalmente una simulazione di boxe: un gioco in stile arcade, che offre al giocatore la stessa prospettiva visiva che avrebbe se si trovasse davvero sul ring. I prodotti Accolade sono distribuiti dalla Electronic Arts.

Electronic Arts
11-49 Station Road, Langley
Berkshire SL3 8YN, UK
(tel. 0044/753/46672)

PIRATI COSMICI

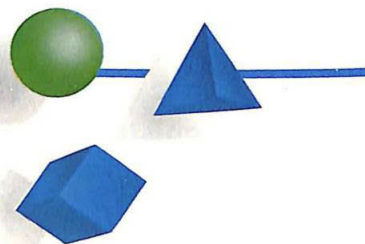
«La pirateria spaziale è una tentazione irresistibile, quando si vedono tutte quelle ricchezze che ci svolazzano attorno nella galassia, aspettando solo che qualcuno se ne impadronisca...».

Non poteva esserci miglior produttore della Outlaw (fuorilegge) per un programma di pirateria cosmica. *Cosmic Pirates* è stato realizzato dalla Zippo Games, un gruppo di programmatori di grande talento che in passato si erano dedicati soprattutto a giochi per le console giapponesi. Ora hanno realizzato questo programma per l'Amiga: il giocatore si trova subito al di là della legge, al soldo della futuristica organizza-

zione corsara "The Council", e viaggia lungo tutta la galassia alla ricerca di astronavi da trasporto con le stive cariche di ricchezze.

Dopo aver sbaragliato le difese avversarie, il giocatore sale a bordo dell'astronave e la guida fino al suo covo. Poi impiegherà il denaro disonestamente guadagnato per comprare armi, motori e schermi di energia migliori e poter affrontare prede sempre più agguerrite e sempre più ricche.

In definitiva, un divertente shoot'em up con una buona presentazione grafica.

Outlaw Productions
The Old Forge
7, Caledonian Road
London N1 9DX, England
(tel. 0044/1/2780751)

Manoscritti: le collaborazioni dei lettori – manoscritti, disegni e/o fotografie– sono benvenute e verranno valutate in vista di una possibile pubblicazione. Commodore Gazette non si assume comunque responsabilità per perdite o danni al materiale. Si prega di allegare una busta affrancata e indirizzata per ogni articolo. Il pagamento per materiale non richiesto viene effettuato solo in seguito all'accettazione da parte della redazione. I contributi editoriali (di qualunque forma) non si restituiscono. Tutta la corrispondenza editoriale, richieste di annunci, problemi di sottoscrizione abbonamenti, di diffusione e con gli inserzionisti, deve essere indirizzata a: Commodore Gazette - Uffici Editoriali - Via Monte Napoleone, 9 - 20121 Milano
La Commodore Gazette è un periodico indipendente non connesso in alcun modo con la Commodore Business Machines e con tutte le sue sussidiarie e affiliate, compresa la Commodore Italiana S.p.A. La Commodore Gazette viene pubblicata mensilmente dalla IHT Gruppo Editoriale, Via Monte Napoleone 9, 20121 Milano. Il costo degli abbonamenti è il seguente : Italia - 12 numeri L. 84.000, 24 numeri L. 168.000, 36 numeri L. 252.000. Estero - Europa L. 120.000 (10 numeri), Americhe, Asia... L. 180.000 (10 numeri). La redazione si adopera per fornire la massima accuratezza negli articoli e nei listati pubblicati. La Commodore Gazette non si assume responsabilità per eventuali danni dovuti a errori od omissioni.

Classified

Software

Scambio software ed esperienze. Possego programmi personali per studenti dell'ITG. Invito tutti i possessori di A500 e altri modelli Amiga, a scrivere o telefonare a: Catalano Tommaso – Via V. Calvanese, 34 – Afragola (NA) – Tel. 8696644

La più vasta biblioteca software in Italia in aggiornamento settimanale con arrivi dai gruppi più famosi L. 4.000 a programma!!! Irripetibile!!! Iscrizione e catalogo gratuiti – Amiga Club Europeo – C.so Trento, 63 – 38086 Pinzolo (TN) – Tel. 0465/51610 (dalle 15.00 alle 19.00)

Vendo programma totocalcio per C-128 – 40/80 colonne, stampe video-carta, schedina con MPS 803, spoglio ecc. Novità: filtro speculare fino al 3° livello. Telefonare a Giuseppe 0924/75275 – ore serali.

Vendo sistema completo GEOS 1.3 con geoCalc, geoFile, geoSpell, Fontpack1, Deskpack1, e Writer's Workshop tutti quanti originali. Solo Roma. Luca – Tel. 06/3765696.

Scambio programmi per C-64 solo su disco. Preferibilmente programmi di utilità grafico - musicali, tool ecc. Massima serietà e velocità. Scrivere a Canè Ciro – Via Arcora Comunale, 7 – 80013 Casalnuovo (NA) - Tel. 081/8421498 – 21 – 22.

Scambio programmi per Commodore Amiga Mandare lista o telefonare ad Alessandro Quartieri – Via Leopardi, 8 – 41012 Carpi (Mo) – Tel. 059/690686 ore pasti.

Vendo software GEOS per Commodore 64 per passaggio a computer superiori. I dischetti sono originali. Vincenzo Canoro – Via S. Giorgio 112 – 55100 Lucca.

Sensazionale: vendo per Amiga e MS-DOS programma "Automatic Mail Post", che archivia clienti per poi registrare vendite per corrispondenza, specialmente di dischi, calcola il il costo dell'operazione, il guadagno, stampa i modelli per effettuare spedizioni con stampa anche delle etichette da attaccare. Il programma viene personalizzato. Danilo D'Agostino – Tel. 085/4152400.

Hardware

Vendo Interfaccia MIDI SIEL a L. 150.000 – Adattatore telematico 6499 a L. 100.000. Tel. 0585/52890 – Claudio.

Vendo Commodore 128, Drive Commodore 1571, registratore, cassette/dischi (programmi originali e non), tutto a L. 900.000 trattabili. Telefonare allo 0131/64888 e chiedere di Corrado (ore pasti).

Vendo Commodore 64, drive 1541, registratore, joystick, TurboDOS, oltre 300 programmi su floppy disk. Tutto a L. 500.000. Telefonare allo 089/951616 – chiedere di Pierluigi.

Vendo Amiga 1000 tastiera e Kickstart (1.1 – 1.2 – 1.3) italiani, monitor 1081, secondo drive (SLIM), tutto perfetto a lire 1.900.000 + Stampante Panasonic 1080 tutto a lire 2.350.000. Telefonare 051/575594 (ore pasti). Serafino Chierici.

Vendo causa passaggio a sistema superiore C-128D, monitor 1901, adattatore telematico 6499, mouse, penna ottica, registratore 1530, cartucce Final Cartridge III freeze frame – doppia spina per registratore, 300 dischi con programmi (anche originali) con manuali a L. 2.000.000 trattabili. Tel. 02/90723291 (Valerio).

Vendo stampante grafica a 4 colori, plotter 1520 Commodore originale. Ha 4 set di caratteri di diverse grandezze. L. 140.000. Affarone! Tel. 0183/400814 oppure 495491.

Vendo Digiview 3.0 hardware e software, potentissimo (Overscan, Halfbrite, Ham, filtri colore per VCR, modo RGB con 4096 colori...) + 20 dischi con i migliori programmi per Amiga + 3 libri e 10 riviste sull'uso e programmazione dello stesso. Tutto in perfetto stato a L. 270.000. Gabriele Zonta – Via Vivai, 9 – 27100 Pavia – Tel. 0382/33921.

CLASSIFIED DELLA COMMODORE GAZETTE È UN MODO ECONOMICO PER INFORMARE LA PIÙ VASTA UTENZA COMMODORE SUI VOSTRI PRODOTTI O SERVIZI. GLI ANNUNCI NON A SCOPO DI LUCRO, INVIATI DA PRIVATI, VENGONO PUBBLICATI GRATUITAMENTE (COMPILARE L'APPOSITA SCHEDA DI SERVIZIO LETTORI).

Quote: 15.000 lire per linea, minimo 4 linee. Aggiungere 5.000 lire per ogni parola in grassetto o 50.000 lire per l'intero annuncio in grassetto.
Condizioni: pagamento anticipato. Vengono accettati assegni e vaglia postali. Gli assegni devono essere intestati a: IHT GRUPPO EDITORIALE s.r.l.
Forma: gli annunci sono soggetti all'approvazione dell'editore e devono essere scritti a macchina o in modo molto chiaro. Una linea equivale a 40 lettere, spazi tra le parole compresi. Pregasi sottolineare le parole che si intendono scrivere in grassetto.
Informazioni generali: gli inserzionisti devono sempre specificare nome e indirizzo completo. Gli annunci appariranno nel primo numero disponibile dopo il ricevimento.

Inviare il materiale a:
IHT GRUPPO EDITORIALE
UFFICI PUBBLICITARI
VIA MONTE NAPOLEONE 9
20121 MILANO

Attenzione: Commodore Gazette non si assume responsabilità in caso di reclami di qualnque natura da parte degli inserzionisti e/o dei lettori. Nessuna responsabilità è altresì accettata per errori e/o omissioni di qualsiasi tipo.

Varie

Per C-64 vendo 8 numeri di "Commodisk" + 8 floppy di programmi tutti con dettagliatissime spiegazioni, 2 "64&128" con floppy e cassetta, un "Elettronica & Computer" con cassetta; regalo 2 cassette giochi! Tel. 0183/400814.

Gruppo di programmatori cerca collaboratore. Requisiti: conoscenza della programmazione in LM su C-64 e/o Amiga, abitante a Bologna o provincia. Per informazioni telefonare allo 051/244688.

D-Mail. Tutto per Commodore. Circuiti integrati per tutta la serie Commodore. Accessori per computer, nuove scatole di montaggio. Offerta speciale: drive OC 118–N (compatibile 1541). Chiedete catalogo gratis. **Delta Computing srl** – Tel. 055/608440 – Fax 055/609227.

Hai un modem? Incontriamoci allora su Skynet, attiva ogni giorno dalle 21 alle 7. Parametri 300 Baud 8/N/I. Telefono 0836/565195. Skynet ti offre tante aree, una stupenda sezione testi, lista altre BBS e una megagalattica area hackers.

Cerco possessori Amiga per scambio informazioni. Telefonatemi per qualunque informazione. Sergio Cavicchioli – Via Donismonda, 31 – 46026 Quistello (Mantova).

Cerco amici con modem e C-64 per scambio informazioni e programmi. Scrivere (col numero di telefono) ad Antonio Presti – Via Enrico Franco, 2 – 90128 Palermo.

Commodore Club

Cerchiamo soci per il "New Commodore Club". Grandi idee, massima serietà, rispondiamo a tutti. New Commodore Club – Via Baracca, 9/11 – 16035 Rapallo (Genova) – Tel. 0185/60829.

È sorto il Free Amiga Club. L'unico club dove tutto è gratis. Per informazioni scrivere o telefonare a Andrea Focardi – Via G. Di Vittorio, 56 – 50015 Grassina (Firenze) – Tel. 055/642046.

INDICE DEGLI INSERZIONISTI

Direzione vendite spazi pubblicitari:

IHT Gruppo Editoriale
Commodore Gazette
Uffici Pubblicitari
Via Monte Napoleone, 9
20121 Milano
Tel. 02/794181 - 799492
792612 - 794122
Telex 334261 IHT I
Telefax 02/784021

Questo indice è da considerarsi come un servizio addizionale. L'Editore non si assume alcuna responsabilità per eventuali errori e/o omissioni.

Indirizzare eventuali lamentele riguardanti gli inserzionisti a:

Commodore Gazette
Uffici Pubblicitari
Relazioni Inserzionisti
Via Monte Napoleone, 9
20121 Milano

Nessuna responsabilità viene altresì assunta dalla Commodore Gazette per eventuali problemi di qualsiasi natura con gli inserzionisti. La responsabilità di quanto pubblicato negli spazi pubblicitari è esclusivamente del committente.

NEL PROSSIMO NUMERO

Gennaio/Febbraio 1989

- **Commodore 64 e 128 -** Nuovi prodotti e novità software per gli intramontabili otto bit della Commodore.

- **L'apprendista programmatore -** I segreti della programmazione dell'Amiga e le nozioni di base: i gadget booleani, i primi elementi di linguaggio Assembly, i parametri della struttura ExecBase.

- **Una guida a GEOS -** Con la descrizione dei box di dialogo e le prove software degli ultimi pacchetti proposti dalla Berkeley, continua e si approfondisce la nostra analisi del sistema operativo GEOS per il C-64/128.

- **Prove stampanti -** Le dettagliate prove hardware di Commodore Gazette hanno ancora come protagoniste le stampanti: sotto esame un'intera famiglia di stampanti OKI, modelli da 9 aghi, 18 aghi, 24 aghi.

- **Corso di programmazione in linguaggio C -** Scopriamo quali sono i principali strumenti di lavoro del programmatore e come farne l'uso migliore: la scelta dell'editor, il compilatore Aztec C della Manx, il source level debugger.

SERVIZIO LETTORI

Questa scheda è valida fino al 28 febbraio 1989

A. Come giudica questo numero di Commodore Gazette?

- ☐ 1. Ottimo
- ☐ 2. Molto buono
- ☐ 3. Buono
- ☐ 4. Discreto
- ☐ 5. Sufficiente
- ☐ 4. Mediocre
- ☐ 6. Insufficiente

B. Quale(i) articolo(i) di questo numero ha apprezzato maggiormente?___

C. Quale(i) articolo(i) di questo numero giudica peggiore(i)?___

D. Quali argomenti dovrebbero essere trattati nei prossimi numeri di Commodore Gazette?___

E. Con quale aggettivo descriverebbe Commodore Gazette?___

F. Quante persone leggono la sua copia di Commodore Gazette?

- ☐ 1. Uno
- ☐ 2. Due
- ☐ 3. Tre
- ☐ 4. Quattro o più

G. Ha dei suggerimenti?___

H. Quale(i) computer utilizza?

- ☐ 1. C-64
- ☐ 2. C-128
- ☐ 3. C-128D
- ☐ 4. Amiga 500
- ☐ 5. Amiga 1000
- ☐ 6. Amiga 2000
- ☐ 7. PC 1
- ☐ 8. PC 10
- ☐ 9. PC 20
- ☐ 10. Altro (specificare)___

I. Quale(i) computer intende acquistare nel futuro?

- ☐ 1. C-64
- ☐ 2. C-128D
- ☐ 3. Amiga 500
- ☐ 4. Amiga 2000
- ☐ 5. PC 1
- ☐ 6. PC 10
- ☐ 7. PC 20
- ☐ 8. Altro (specificare)___

L. È un acquirente dei libri della IHT? Se sì come li giudica?___

M. Quali altre riviste (sia d'informatica che non) legge abitualmente?___

N. Indichi quali sono i suoi maggiori interessi

- ☐ 1. Videoregistrazione
- ☐ 2. Hi-Fi
- ☐ 3. Fotografia
- ☐ 4. Automobili
- ☐ 5. Sport
- ☐ 6. Viaggi

Nome ___
Cognome ___
Indirizzo ___
Città ___
Prov. ___ **C.a.p.** ___
Età ___ **Professione** ___

COMMODORE GAZETTE **dicembre 1988**

SCHEDA ORDINAZIONE LIBRI

Con il presente tagliando desidero ordinare i seguenti libri:

- ☐ L'Amiga (Michael Boom) .... L. 60.000
- ☐ Il Manuale dell'AmigaDOS (Commodore-Amiga) .... L. 60.000
- ☐ Programmare l'Amiga Vol. I (Eugene P. Mortimore) .... uscita: maggio '89
- ☐ Programmare l'Amiga Vol. II (Eugene P. Mortimore) .... uscita: febbraio '89
- ☐ Guida ufficiale alla programmazione di GEOS (Berkeley Softworks) .... L. 64.000
- ☐ Flight Simulator Co-pilot (Charles Gulick) .... L. 30.000
- ☐ Volare con Flight Simulator (Charles Gulick) .... L. 45.000
- ☐ Le Mille luci di Hollywood (David Chell) .... L. 39.900

Pagherò in contrassegno al postino la somma di L.............................+ spese postali (L. 5.000 per volume)

Nome ___
Cognome ___
Indirizzo ___
Città ___
Prov. ___ **C.a.p.** ___ **Tel.** ___
Firma ___

COMMODORE GAZETTE **dicembre 1988**

☐ Desidero inserire gratuitamente un mio annuncio nella rubrica CLASSIFIED (solo per i privati e per gli annunci non a scopo di lucro).

☐ Desidero inserire un mio annuncio nella rubrica CLASSIFIED. Allego assegno bancario o circolare o postale oppure fotocopia della ricevuta del vaglia postale per un totale di L. ______________________ . Il mio codice fiscale o partita IVA (per le aziende) è il seguente:______________________.

Attenzione: perché un annuncio venga accettato è necessario che sia stato compilato anche il questionario presente sull'altro lato di questo tagliando.

TESTO:______________________

____

____

____

Inserire all'interno di una busta affrancata e spedire a:

Commodore Gazette
Servizio Lettori
Via Monte Napoleone, 9
20121 Milano

- - - - -✂- -

Inserire all'interno di una busta affrancata e spedire a:

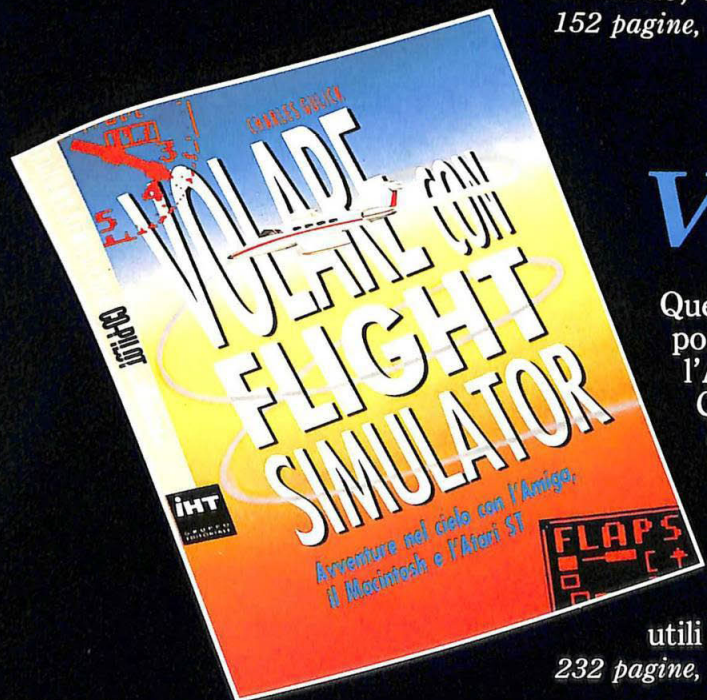
IHT Gruppo Editoriale
Divisione Libri
Via Monte Napoleone, 9
20121 Milano

UN COMPUTER, UN LIBRO...
...LE ALI

FLIGHT SIMULATOR CO-PILOT

Questo volume è un vero "istruttore di volo" per coloro che possiedono il programma Flight Simulator della Microsoft per i PC IBM e compatibili, oppure Flight Simulator II della SubLogic per Commodore 64, Apple II, Atari 800 XL e XE.

Non è un semplice compendio di comandi da ricordare a memoria, ma un brillante compagno d'avventura, scritto con estro e ironia. Un libro da sfogliare con il computer acceso, per imparare passo dopo passo i principi del volo "rettilineo e livellato", le procedure di decollo e di atterraggio, del volo strumentale e notturno, e del volo acrobatico.

152 pagine, L. 30.000

VOLARE CON FLIGHT SIMULATOR

Questo volume si rivolge a tutti gli utenti che possiedono la versione di Flight Simulator per l'Amiga, l'Atari ST o l'Apple Macintosh.

Con uno stile spigliato, ma con grande attenzione a ogni dettaglio, l'autore conduce l'allievo pilota attraverso le varie fasi di un corso di volo che si concluderà con un primo approccio al volo acrobatico. Il testo è completato da dettagliate cartine di rotta, schemi che illustrano l'uso dei comandi, parametri di volo ideali e procedure utili nelle più svariate situazioni di volo.

232 pagine, L. 45.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127
IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I
Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

REGALA IL CINEMA